

NAVAL POSTGRADUATE SCHOOL Monterey, California

AD-A238 888



DTIC
ELECTE
JUL 22 1991
S C D

THESIS

DESIGN AND IMPLEMENTATION OF A PROTOTYPE
MICROCOMPUTER DATABASE MANAGEMENT SYSTEM
FOR THE STANDARDIZATION OF
DATA ELEMENTS FOR THE
THE DEPARTMENT OF DEFENSE

by

John S. Bacheller

September 1990

Thesis Advisor:

Daniel R. Dolk

Approved for public release; distribution is unlimited.

91-05545



91 7 18 025

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED			1b. RESTRICTIVE MARKINGS		
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution is unlimited.		
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE			5. MONITORING ORGANIZATION REPORT NUMBER(S)		
4. PERFORMING ORGANIZATION REPORT NUMBER(S)			5. MONITORING ORGANIZATION REPORT NUMBER(S)		
6a. NAME OF PERFORMING ORGANIZATION Naval Postgraduate School		6b. OFFICE SYMBOL (If applicable) 55		7a. NAME OF MONITORING ORGANIZATION Naval Postgraduate School	
6c. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000			7b. ADDRESS (City, State, and ZIP Code) Monterey, CA 93943-5000		
8a. NAME OF FUNDING/SPONSORING ORGANIZATION		8b. OFFICE SYMBOL (If applicable)		9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
8c. ADDRESS (City, State, and ZIP Code)			10. SOURCE OF FUNDING NUMBERS		
			Program Element No	Project No	Task No
					Work Unit Accession Number
11. TITLE (Include Security Classification) DESIGN AND IMPLEMENTATION OF A PROTOTYPE MICROCOMPUTER DATABASE MANAGEMENT SYSTEM FOR THE STANDARDIZATION OF DATA ELEMENTS FOR THE DEPARTMENT OF DEFENSE					
12. PERSONAL AUTHOR(S) Bacheller, John S.					
13a. TYPE OF REPORT Master's Thesis		13b. TIME COVERED From To		14. DATE OF REPORT (year, month, day) September, 1990	
				15. PAGE COUNT 247	
16. SUPPLEMENTARY NOTATION The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.					
17. COSATI CODES			18. SUBJECT TERMS (continue on reverse if necessary and identify by block number)		
FIELD	GROUP	SUBGROUP	Data Management Standardization, Data Element Dictionary, Data Element Standardization, Object Oriented, Logical Design, Application Design.		
19. ABSTRACT (continue on reverse if necessary and identify by block number) The need for data management standardization has been clearly identified by the Office of the Secretary of Defense as a means to realize savings in the \$9 billion spent annually on information technology in DOD and to further the sharing of information. This thesis discusses the importance of data element standardization as a foundation for standardizing Management Information Systems within DOD. Moreover, this thesis identifies the data requirements, functional requirements, logical database design and application design for a prototype microcomputer dictionary system for standardizing, storing, updating and viewing data elements, the lowest level in the hierarchy of metadata. This prototype dictionary is then implemented in a powerful relational database management system, Paradox 3.0. This system will help developers within DOD to build, store, track and use standard data elements.					
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT <input checked="" type="checkbox"/> UNCLASSIFIED/UNLIMITED <input type="checkbox"/> SAME AS REPORT <input type="checkbox"/> LIMIT USERS			21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED		
22a. NAME OF RESPONSIBLE INDIVIDUAL Daniel R. Dolk			22b. TELEPHONE (Include Area code) (408) 646-2260		22c. OFFICE SYMBOL DK54

Approved for public release; distribution is unlimited.

DESIGN AND IMPLEMENTATION OF A PROTOTYPE
MICROCOMPUTER DATABASE MANAGEMENT SYSTEM FOR THE
STANDARDIZATION OF DATA ELEMENTS FOR THE
DEPARTMENT OF DEFENSE

by

John S. Bacheller
Captain, United States Marine Corps
B.S., University of Southern California, 1982

Submitted in partial fulfillment
of the requirements for the degree of

MASTER OF SCIENCE IN INFORMATION SYSTEMS

from the

NAVAL POSTGRADUATE SCHOOL

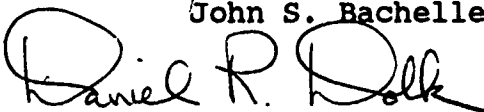
September 1990

Author:

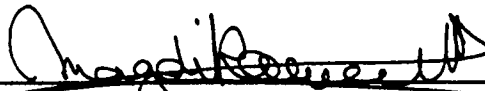


John S. Bacheller

Approved by:



Daniel R. Dolk, Thesis Advisor



Magdi N. Kamel, Second Reader



David R. Whipple, Chairman,
Department of Administrative Sciences

ABSTRACT

The need for data management standardization has been clearly identified by the Office of the Secretary of Defense as a means to realize savings in the \$9 billion spent annually on information technology in DOD and to further the sharing of information.

This thesis discusses the importance of data element standardization as a foundation for standardizing Management Information Systems within DOD. Moreover, this thesis identifies the data requirements, functional requirements, logical database design and the application design for a prototype microcomputer dictionary system for standardizing, storing, updating and viewing data elements, the lowest level in the hierarchy of metadata. This prototype dictionary is then implemented in a powerful relational database management system, Paradox 3.0.

This system will help developers within DOD to build, store, track and use standard data elements.

Accession For	
NTIS GRA&I	<input checked="" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Avail and/or	
Dist	Special
A-1	

TABLE OF CONTENTS

I. DEFINITION PHASE	1
A. BACKGROUND	1
B. PROBLEM DEFINITION	4
C. PROJECT SCOPE	5
1. Technical Feasibility	6
2. Economical Feasibility	6
3. Political Feasibility	6
D. METHODOLOGY	7
II. REQUIREMENTS PHASE	9
A. DATA REQUIREMENTS	9
1. The Concept of Data and the Data Element	9
2. The Structure of Data: "Object Oriented Methodology"	15
B. FUNCTIONAL COMPONENT REQUIREMENTS	19
1. Update Mechanisms	20
2. Display Mechanisms	22
3. Control Mechanisms	25
4. Summary	25
III. DESIGN PHASE	26

A. LOGICAL DATABASE DESIGN	26
B. APPLICATION DESIGN	30
1. Application Functions Scope	31
2. Menu Hierarchy and Materializations	33
a. Reference Element Option	33
b. Data Element Option	37
c. Alias Element Option	40
d. Class Word Option	43
e. Prime Word Option	45
f. Unique Queries Option	46
g. Help Option	47
h. Leave Application Option	47
5. Summary	47
IV. IMPLEMENTATION PHASE	48
A. SELECTION OF PARADOX 3.0	48
B. CONSTRUCTION OF THE PROTOTYPE USING PARADOX 3.0	49
C. PROBLEMS AND BENEFITS OF PARADOX 3.0 IN IMPLEMENTATION	56
D. USERS MANUAL AND SCRIPT CODE	58
V. CONCLUSION AND RECOMMENDATIONS	59
APPENDIX A: PRIME WORD LIST	61
APPENDIX B: CLASS WORD LIST	67

APPENDIX C: AR 25-9 NAMING CONVENTIONS	69
APPENDIX D: OBJECT DIAGRAMS	72
APPENDIX E: OBJECT PROPERTY DOMAIN DEFINITION	76
APPENDIX F: OBJECT RELATIONAL NARRATIVE	116
APPENDIX G: FORM/REPORT MATERIALIZATIONS	119
APPENDIX H: USER'S MANUAL	162
APPENDIX I: MAIN SCRIPT PAL CODE	183
LIST OF REFERENCES	237
INITIAL DISTRIBUTION LIST	239

ACKNOWLEDGEMENT

This thesis is dedicated with love to my wife, Maggie, and my wild children, Leigh and Stephen.

Special thanks to Bruce Haberkamp, Maryjo Matera, and Maj Lasher at ODISC4 and LtCol Phil Olson at CIM. Also, special thanks to Glenda Hurley, Becky Harris, Jim Glymph and Jim Pypher at DMD, Ft. Belvoir. Cheers!

I. DEFINITION PHASE

A. BACKGROUND

The need for data management standardization has been clearly identified by the Office of the Secretary of Defense (OSD). Savings in both the \$9 billion spent annually on information technology and in Department of Defense (DOD) business-related areas (that these systems support) can be realized through standardization [Ref. 1]. OSD's general goals are:

1. Consolidate multiple systems that meet the same functional requirements.
2. Reduce unnecessary redundancy.
3. Develop common data requirements and formats.

As a result of the implementation of Defense Management Report Decision (DMRD) 925, DOD Corporate Information Management (CIM) was established in October 1989, as the agency directed to "enhance the availability and standardization of information in common areas and provide for the development of integrated Management Information Systems (MIS) [Ref. 2]."

CIM, in its infancy, is working closely with the Department of the Army's Office of the Director of Information Systems for Command, Control, Communications and Computers

(ODISC4) to promulgate data management standardization. ODISC4 is the senior policy official for data management in the Army (see Figure 1.1). The Army Data Management and

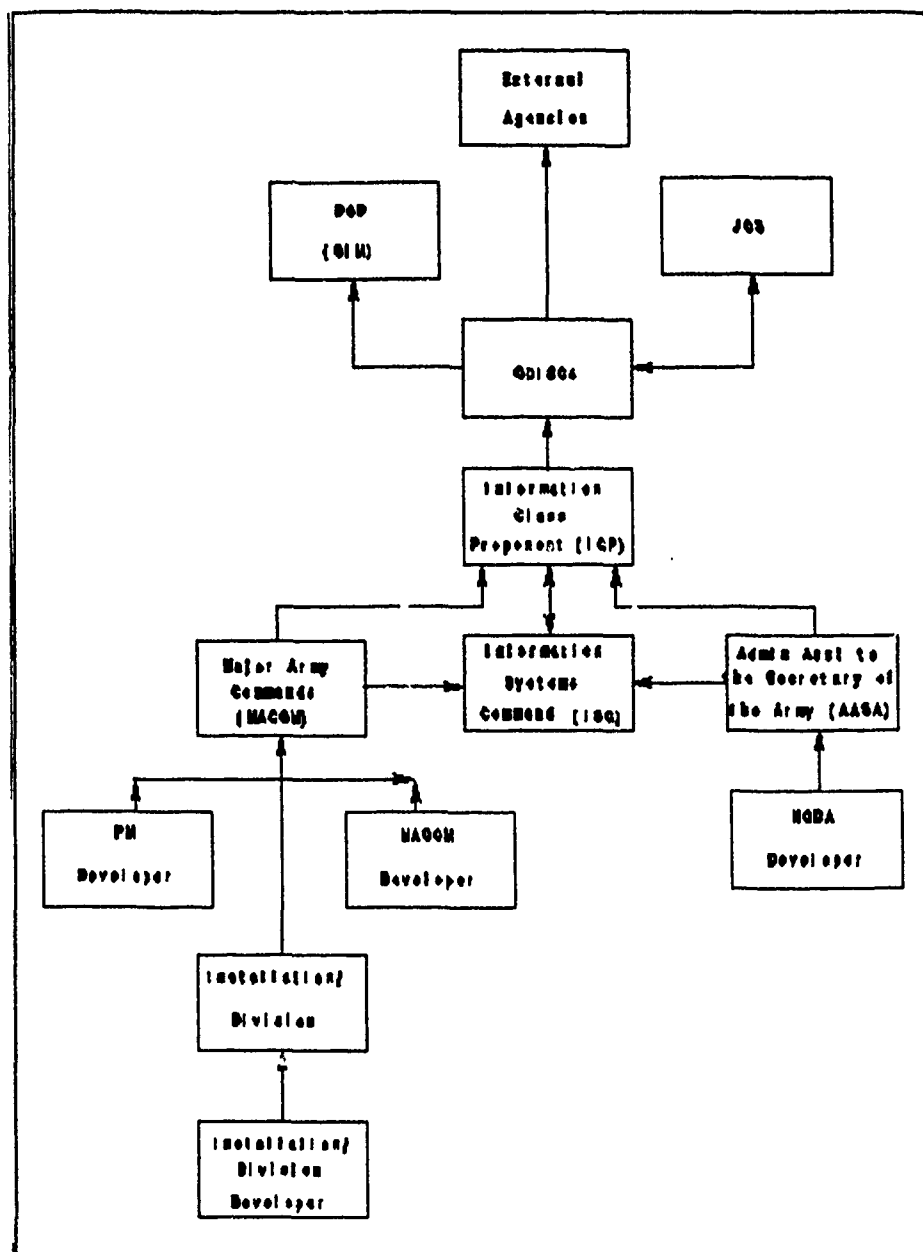


Figure 1.1. Data Standardization Chain-of-command.

Standards Program, Army Regulation 25-9 (AR 25-9) published on September 25, 1989, provides general guidelines and specific

rules and responsibilities for data element standards [Ref. 3].

The data element is the lowest level in the hierarchy of metadata, or data that describes data (see Figure 1.2) [Ref. 4]. Because it is the lowest level, standardization is crucial, and therefore a top priority in the eyes of CIM and ODISC4. This focus on the data element as the foundation for standardization has led to creation of the Army Data Dictionary (ADD) Automated Dictionary Support System (ADSS).

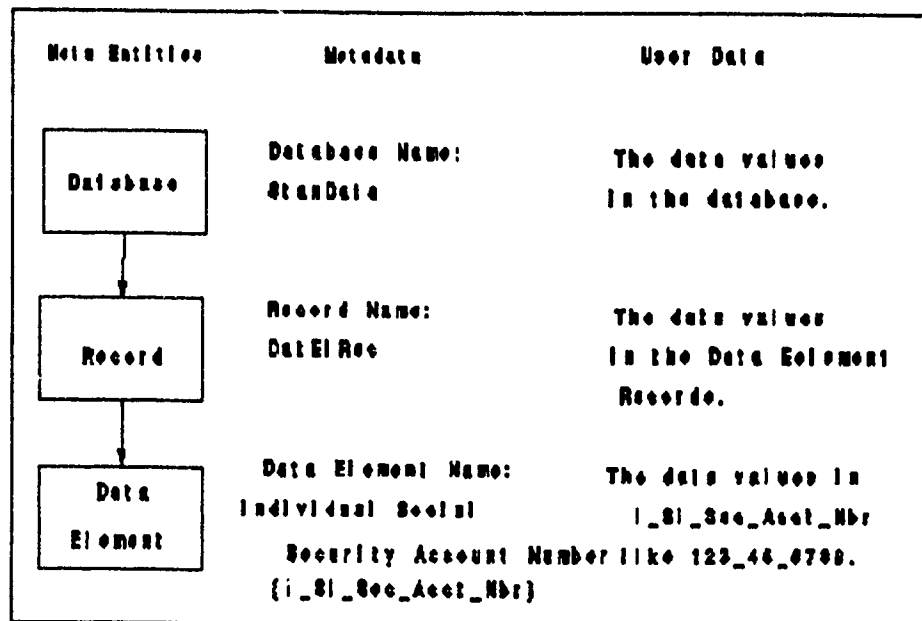


Figure 1.2. Data Element as Metadata.

This system resides on an IBM 90370 minicomputer and is managed by the Data Management Directorate (DMD), a subagency of the Army Information Systems Command (ISC). The ADD is a

repository of data elements and the building blocks from which they are made [Ref. 5]. The purpose of the ADD ADSS is to ensure information sharing among organizational elements across functional lines. It allows the Information Class Proponents (ICP: the owners of a data element according to information class), organization data administrators and system developers to capture, query, maintain and approve Army standard elements. It is accessible through the:

1. Army Management Support Network (AMSNET).
2. Management Support Network (MASNET).

Comments, candidate elements and element approvals or disapprovals can be uploaded to the system via these networks. Currently, however, the data elements cannot be downloaded via network (dial-up line) to the ICP's, data administrators and developers. Though the information is available in 'hard' copy and nine track tape, there are no dedicated subsystems (minicomputer and microcomputer) with a relational Database Management System (DBMS) that can be used at lower levels as a data element dictionary and desktop glossary.

B. PROBLEM DEFINITION

The problem is to create a prototype data element dictionary that will allow ICP's, data administrators and

developers to manipulate, maintain and view standard data elements and the processes which support their lifecycle from naming convention to definition and approval. Based on the emphasis of cost-savings through standardization, this prototype system must satisfy the following constraints:

1. Use current in-house 80286 central processing unit (cpu) based microcomputers.
2. Use no more than 640 kilobytes (K) of Random Access Memory (RAM).
3. Use an off-the-shelf DBMS, preferably a relational DBMS.

C. PROJECT SCOPE

The scope of this thesis is confined to the creation of a prototype data element dictionary within the given constraints. This dictionary will employ the logical model and physical table structure of the ADD ADSS to the greatest extent possible to encourage future considerations of an indirect (disk or tape media) or direct (dedicated/dial-up line) data download capability. However, this system will be built on the premise of manual data entry. Also, the ADD ADSS is written in Standard Query Language (SQL) Cobol and few commercial microcomputer DBMS's offer complete SQL

capability within RAM limits. Therefore, differences in the two systems will exist.

1. Technical Feasibility

Both the hardware and the software needed to develop this prototype are available. The software selected as the relational DBMS is Borland's Paradox 3.0. Paradox uses its own Programming Application Language and has future growth potential in soon-to-be-released Paradox SQL and a Compiler (TSR Corporation, New York). One minor issue is that data field lengths are limited to 250 characters which falls short of a few fields described in AR 25-9.

2. Economical Feasibility

Overall, savings in the current budget of \$9 billion as well as the out-year savings are the "drivers" of this standardization issue. For this thesis, use of existing facilities and equipment as well as "research-oriented" labor will help minimize costs. User training costs will be kept at a minimum because the prototype will be menu driven and require little dedicated training.

3. Political Feasibility

In an environment where "budget deficit reduction" is the watch-word and consolidation and centralization of functions is the trend, development of a prototype system to aid standardization can only be viewed as a positive

impetus toward information sharing, communication, and resulting cost savings. The CIM/ODISC4 mission needs to be supported by all services. This dictionary system positively supports the CIM/ODISC4 mission.

D. METHODOLOGY

The methodology of this thesis will center on the following:

1. Provide CIM and Army background, define the problem and describe a proposed solution.
2. Using an object-oriented approach, determine and develop user requirements and relational database design for a passive data element dictionary. The object-oriented approach involves creating data objects from the data required by the user for inclusion in the system. This approach is described in detail in Chapter II.
3. Implement the relational design in a prototype system. Provide a User's Manual for the prototype system.

The structure of this thesis will mirror the methodology. Chapter II describes the system requirements, including the definition and structure of the user's data as well as the functional components of the system (update, display and control mechanisms). Chapter III develops a relational design of the dictionary. Chapter IV presents an implementation of the design, including a User's Manual.

Chapter V summarizes conclusions and suggests further enhancements to the prototype.

II. REQUIREMENTS PHASE

The purpose of the requirements phase for a database application is to determine, from user input, what data will be used and how that data will be manipulated.

Specifically, user requirements definition involves satisfying two primary goals: (1) identification of data requirements (the concept and the structure of data that represents the user's environment), and (2) identification of the functional components (update, display and control mechanisms) that the applications will use. The following sections will address each of these goals.

A. DATA REQUIREMENTS

1. The Concept of Data and the Data Element

The term "data element" has been used loosely up to this point. Because this prototype is essentially a data element dictionary, it hinges on the concept of data and the data element as defined and standardized in AR 25-9.

Data are the basic units of information in information systems and are represented as raw numbers, words or codes. An item or instance of data is called a

data value. There are two types of data values, qualitative and quantitative. Qualitative data consists of data values that represent some aspect of a "thing" and are construed as literal data (like words in a paragraph) or data code (symbology used to represent literal data like social security number). Quantitative data are numerical expressions of data in real number or integer format. For example, the word "name" is qualitative, identifying or qualifying an object. On the other hand, the word "length" is quantitative, providing a quantified measurement.

A data element is a named piece of data that is of interest to a person or organization. It describes or defines an attribute or quality of an entity (person, place, thing, object, concept or event) or relationship [Ref. 6]. It must be unambiguously defined, logically consistent, and possess a homogeneous domain of associated values. An example of a standardized data element is "Individual Social Security Account Number." By using the word "Individual" instead of a more qualified term like "Officer's" or "Spouse's," redundancy in capturing information has been avoided. For example, if qualified terms were used, the database might have at least five separate data elements dedicated to identifying social security number:

1. Officer's Social Security Number

2. Warrant Officer's Social Security Number
3. Enlisted's Social Security Number
4. Spouse's Social Security Number
5. Child's Social Security Number

Data elements one through five each contain the same type of data but for different types of individuals. If the qualification is raised one level to make the individuals generic, then one data element may be used instead of five or more (in the case of more, consider a person with ten children, etc.). Using the "single" data element for social security number, a relationship or qualification can be generated by associating another data element with it (see Figure 2.1) [Ref. 7].

Standardized data elements should not contain relationships or dependencies. Where these dependencies exist, data elements can be raised one level or more into a more generic data description level in order to "singularize" it. Thus, an individual either has a social security number or he does not; and, as data administrators or developers, we should not have to manage a myriad of differently-named data elements that describe the same thing, in this case social security number. AR 25-9 provides a set of rules and constraints for naming, structuring, and defining these data elements [Ref. 8].

These rules prevent redundancy, minimize the amount of data elements needed, and provide guidelines to the developer for creation of new data elements that may be needed in a new system.

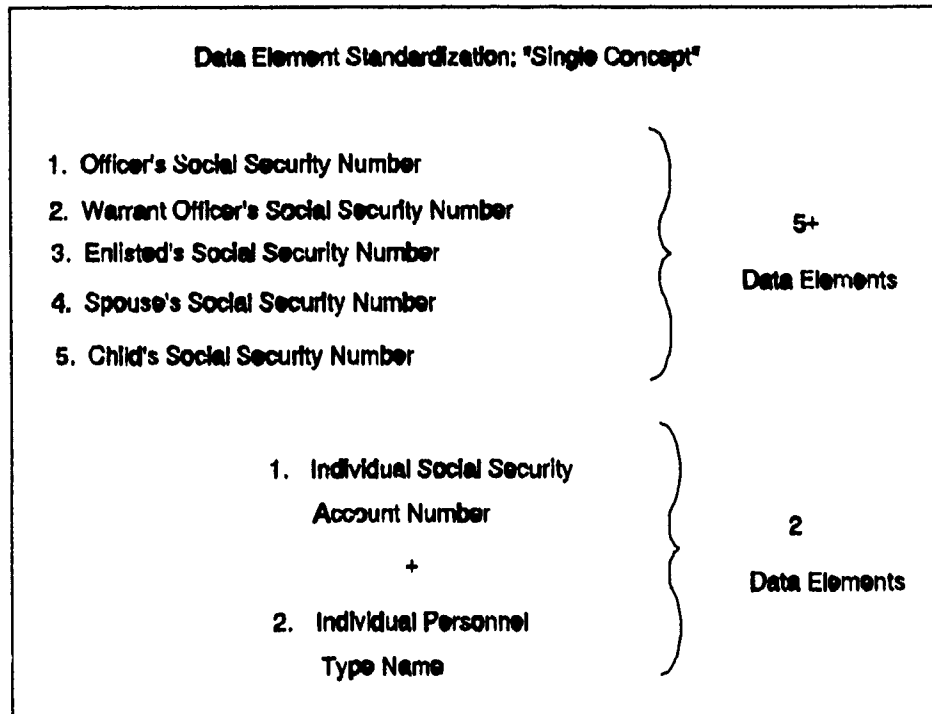


Figure 2.1. Single Data Element Concept.

a. Data Element Naming and Structure

Data elements consist of a name, attributes (for description), and a general or specific domain. A data element acquires organizational context through a "prime term" and inherits its structure and domain of values from a "reference element." AR 25-9 emphasizes that

standardization is gained through the structure of the data element, not through its use.

(1) *Naming.* A data element name is constructed by adding a prime term to a reference element (see Figure 2.2). The prime term has a prime word (required) and up to

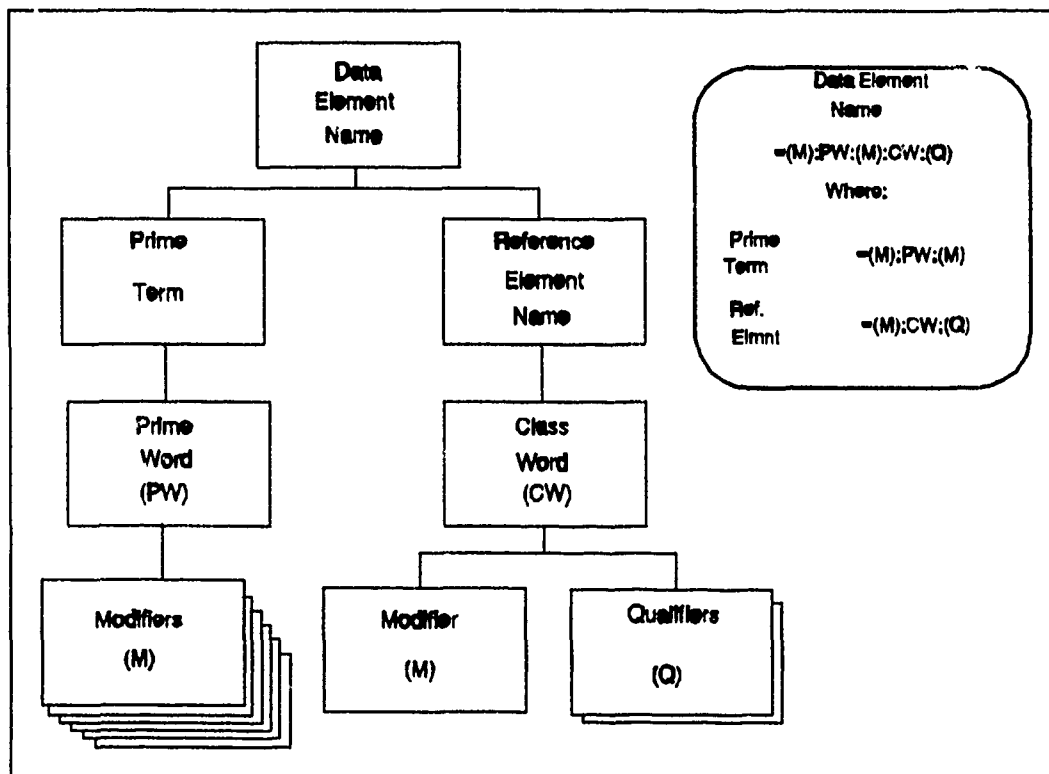


Figure 2.2. Data Element Naming.

six optional modifiers. Prime words come from a finite set within each Army data subject-area. Positionally, the prime word can reside anywhere within the prime term. See Appendix A for a list of prime words and their associated

data subject-areas. A modifier helps refine or render a unique name for a data element, and it cannot be a prime word or a class word. For example, in the data element "Individual Social Security Account Number," the prime term is "Individual Social Security Account." The prime word is "Individual" and comes from the Army data subject-area of Personnel. The modifiers in this case are "Social Security Account" and serve to uniquely identify the prime word.

The reference element has an optional modifier, a class word (required), and two optional qualifiers (in that order). The modifier plays the same role as described above. The class word specifies the type of information contained in a set of data values. See Appendix B for a list of class words. Qualifiers "further describe a characteristic of the information within a common set of data values" [Ref. 9]. For example, in the data element "Individual Social Security Account Number," the reference element is "Number." See Appendix C for a detailed list of naming conventions from AR 25-9.

(2) *Structure.* The data element derives its structure from the reference element. The data element will either take on the domain of the reference element or a subset of it. Thus, if a reference element is a qualitative

data value type, then the data element that is derived from it will also be of qualitative data value type. In the example "Individual Social Security Account Number," the reference element is "Number" which is qualitative data in the form of data code. As a result, the data element is qualitative data in the form of data code and adopts the domain (or a subset) of "Number" (see Figure 2.3). A specific example of the data element domain is: "a specific domain of nine characters comprised of the characters 0-9."

To associate nonstandard data element names used in existing information systems with standardized data elements, the term "data element alias" will be used. The data element alias will be associated with its host system and location. When associated with a data element, it will reflect the data value type (qualitative or quantitative) of the data element. As nonstandardized information systems grow obsolete and/or are phased-out, data element aliases will be eliminated.

2. The Structure of Data: "Object Oriented Methodology"

An "object oriented" methodology will be used to determine and present the user's data requirements. According to Kroenke and Dolan, an object is defined as "a named collection of properties that sufficiently describes an entity in the user's work environment [Ref. 10]." An

"entity" is defined as an independent unit that owns its own elements. A "property" is a characteristic of the entity

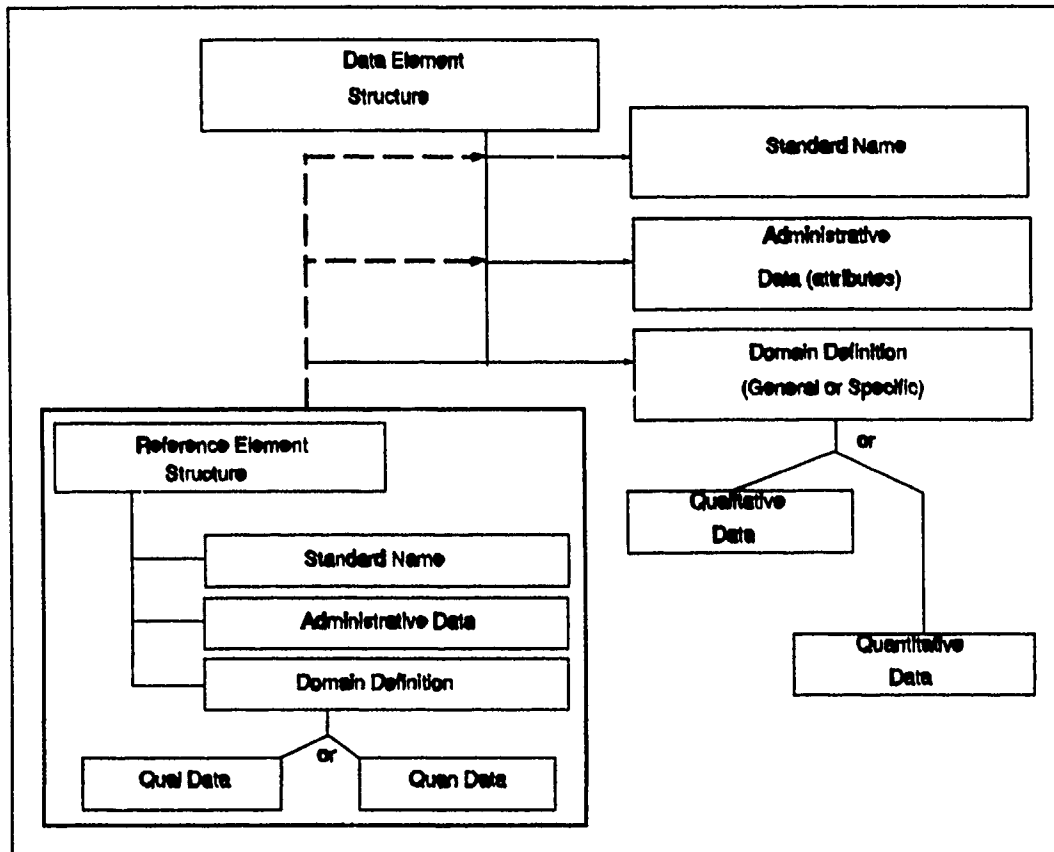


Figure 2.3. Data Element Structure.

(an item that is "owned"). For example, DATA ELEMENT (note the capitalization) is the object and Data Element Number, Data Element Name, and Data Element Alias Name may be properties. Properties can be singular, as in the case of Data Element Number, where a DATA ELEMENT has just one number or multi-valued (mv), as in the case of DATA ELEMENT ALIAS NAME, where a DATA ELEMENT can have more than one alias name. A property may also be an object, with

properties of its own. In the last example, DATA ELEMENT ALIAS NAME is really a multi-valued object property.

Entities, objects, and properties can be combined into an **object diagram** that provides a visual presentation of the data within the user's environment. Boxes are used to represent an object and an object's properties are shown within the box. Figure 2.4 shows an object diagram of the DATA ELEMENT example.

The objects used in the Data Element Dictionary prototype closely follow those used in the ADD ADSS.

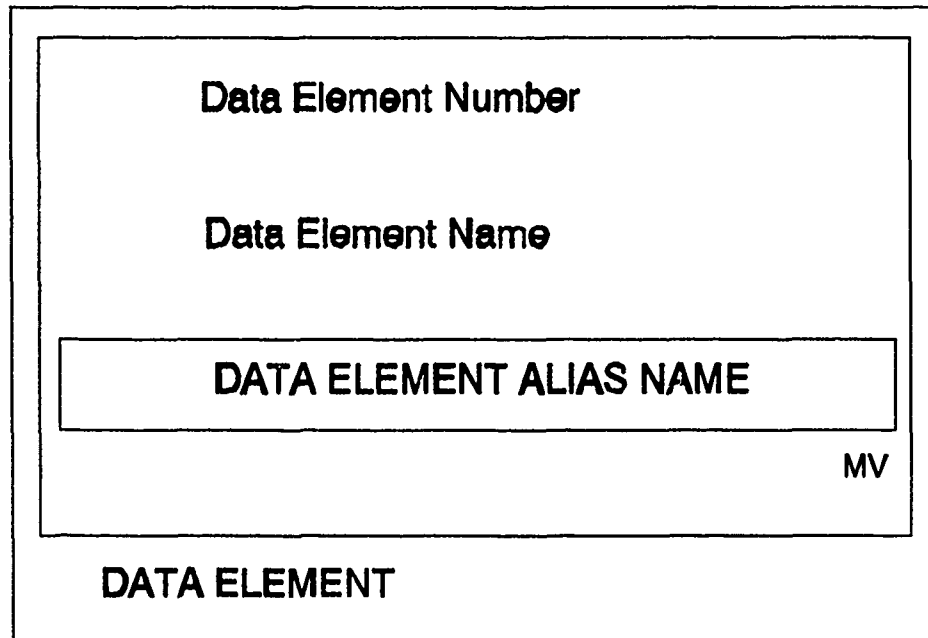


Figure 2.4. Object Diagram.

Thirteen of the eighteen ADD ADSS objects will be used and retain the same name except where limited by the eight

character filename constraint enforced by DOS. The thirteen objects are as follows:

1. APWTRELM = Reference Element - A reference element's attributes.
2. APWTREQD = Reference Element Data Value Number - A reference element's quantitative data values.
3. APWTREFD = Reference Element Data Item - A reference element's qualitative data values.
4. APWTCWOR = Class Word - Class Words and descriptions.
5. APWTDE = Data Element - A data element's attributes.
6. APWTDEQD = Data Element Data Value Number - A data element's quantitative data values.
7. APWTDEDI = Data Element Data Item - A data element's qualitative data values.
8. APWTAL = Alias Element - An alias' attributes.
9. APWTALQD = Alias Element Data Value Number - An alias' quantitative data values.
10. APWTALDI = Alias Element Data Item - An alias' qualitative data values.
11. APWTSS = Alias Element Host System - Alias' Host system information.
12. APWTPWRD = Prime word and related data subject-area name.
13. APWTIC = Information class, and proponent.

Object diagrams can be viewed in Appendix D. The description and domain definition of the object properties can be viewed in Appendix E.

Once the data requirements have been determined by applying user needs and wants to the object-oriented methodology, the functional components of the dictionary, i.e., how the data will be manipulated, must be determined.

B. FUNCTIONAL COMPONENT REQUIREMENTS

Data, data flows, and data processes as well as user interaction characterize the update, display and control mechanisms that are needed in a system. Data Flow Diagrams (DFD) reveal data and its processes and show users where the user fits-in.

The Yourdon methodology [Ref. 11] was used to develop a "Context Diagram" (top level diagram) to provide a general picture of the proposed system (see Figure 2.5). In the proposed system, the user adds and updates data (update in this case is defined as edit and delete) whose general structure was shown in the object diagrams. Also, the user requests reports (in hard-copy) or queries the system for a "view" (Screen Display) of data. The system interacts with the database to perform these functions.

Specific update, display and control mechanisms are as follows:

1. Update Mechanisms

a. Add new Reference Element, Data Element, Alias, Class Word, or Prime word

(1) Inputs. From the user via keyboard.

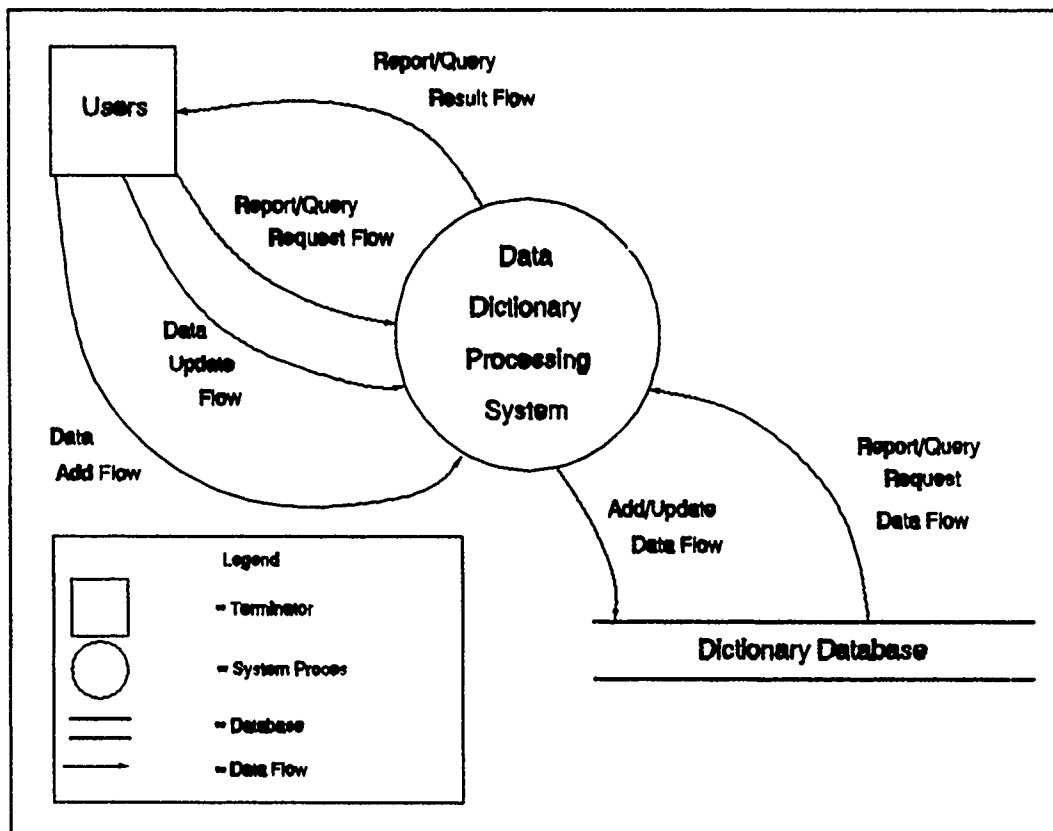


Figure 2.5. Data Flow Diagram.

(2) Outputs. New object instances.

(3) Processing notes. General: Must follow Naming Conventions in AR 25-9. Data value type (Qualitative or Quantitative) will be determined by the Reference Element. Data values associated with Reference, Data, and

Alias Elements may be added also. Reference Element:
Reference Element Number is required and is built from
existing Class Words. Data Element: Data Element Number is
required is built from existing and approved Reference
Elements and Prime Words. Alias: Alias Number is required.
Class Word: Class Word Name is required. Prime Word:
Prime Word Name is required.

(4) Volume of records (initial). Reference
Elements will have approximately 70. Data Elements will
have approximately 50. Alias Element volume is unknown.
Class Words will have approximately 39. Prime Words will
have approximately 250.

**b. Edit existing Reference Element, Data Element,
Alias, Class Word or Prime Word**

(1) Inputs. From the user via keyboard.
(2) Outputs. Revised object instances.
(3) Processing notes. Reference, Data, Alias
Element Numbers cannot be revised or deleted. All other
fields including data values may be changed.

**c. Delete existing Reference Element, Data Element,
Alias, Class Word or Prime Word**

(1) Inputs. From the user via keyboard.

(2) *Outputs.* Removal of object instances.

(3) *Processing notes.* All "one-to-many" relationships existing on the deletion forms (like data values) must be first deleted, then deletion of the associated Reference, Data, or Alias Element is permitted.

2. Display Mechanisms

a. View Options

(1) *View Lists.* View a list of Reference Element Numbers and names, Data Element Numbers and Names, Alias Numbers and Names, Class Word Names, and Prime Word Names and subject-area. Sources for each are as follows:

1. Reference Element: Reference Element Object
(APWTRELM)
2. Data Element: Data Element Object
(APWTDE)
3. Alias Element: Alias Element Object
(APWTAL)
4. Class Word: Class Word Object
(APWTCWOR)
5. Prime Word: Prime Word Object
(APWTPWOR)

(2) *View Detail.* View detailed and unique Reference Element, Data Element, Alias Element, Class Word, and Prime Word object instances on data forms within the Edit option. Sources for each are as follows:

1. Reference Element
 - Reference Element Object (APWTRELM)
 - Reference Element Data Item Object (APWTREFD)
 - Reference Element Data Value Number Object (APWTREQD)
2. Data Element
 - Data Element Object (APWTDE)
 - Data Element Data Item Object (APWTDEDI)
 - Data Element Data Value Number Object (APWTDEQD)
3. Alias Element
 - Alias Element Object (APWTAL)
 - Alias Element Data Item Object (APWTALDI)
 - Alias Element Data Value Number Object (APWTALQD)
 - Alias Element Host System Object (APWTSS)
4. Class Word
 - Class Word Object (APWTCWOR)
5. Prime Word
 - Prime Word Object (APWTPWOR)

b. Printed Report Options

(1) *Print Lists (First Type)*. A printed list report will be generated on Reference Element, Data Element, and Alias Element object instance Names and numbers. Sources for each are the same as those listed in 'a,' bullet one.

(2) *Print Lists (Second Type)*. A printed list report will be generated on Class Word Names and on Prime Word Names and Subject-areas. Sources for each are the same as those listed in 'a,' bullet one.

(3) *Print Detail (First Type)*. Detailed printed reports will be generated on Reference Element, Data Element, Alias Element, Class Words, and Prime Word object instances and will be selected by the object's unique number. Sources are the same as those listed in paragraph 'a,' bullet two.

(4) *Print Detail (Second Type)*. Detailed printed reports will be generated on the data values associated with a Reference Element, Data Element, and Alias Element object instances and be selected by the object's unique number. Data values are either qualitative or quantitative. Sources are as follows:

1. Qualitative Reference Element
Reference Element Object (APWTRELM)
Reference Element Data Item Object (APWTREFD)
2. Quantitative Reference Element
Reference Element Object (APWTRELM)
Reference Element Data Value Number Object (APWTREQD)
3. Qualitative Data Element
Data Element Object (APWTDE)
Data Element Data Item Object (APWTDEDI)
4. Quantitative Data Element
Data Element Object (APWTDE)
Data Element Data Value Number Object (APWTDEQD)
5. Qualitative Alias Element
Alias Element Object (APWTAL)
Alias Element Data Item Object (APWTALDI)

6. Quantitative Alias Element

Alias Element Object

(APWTAL)

Alias Element Data Value Number Object

(APWTALQD)

3. Control Mechanisms

a. Password Control

A password system will be used in connection with the initial introduction screen. Only users with a valid password will have access to applications beyond that.

b. Form Control

All objects with their associated update and display mechanisms will be controlled via form. Form fields will be marked and tailored to receive data in a particular format. For example, because a Reference Element is built from existing Class Words, only existing Class Words in their character-string format will be allowed in that specific field.

4. Summary

In this chapter, user needs were converted into Data Requirements, using an object-oriented methodology, and Functional Component Requirements were converted into update, display and control mechanisms. These requirements will act as the base from which the Design Phase, Logical Design and Application Design, will be built.

III. DESIGN PHASE

Whereas the Requirements Phase involves determination of user needs, the Design Phase translates these requirements into: (1) Logical Database Design consisting of a transformation of the user's data objects into a relational diagram, a description of relationships between objects and relationship constraints and (2) Application Design, including the scope of the functions of the application, a menu hierarchy, and materializations of the menus, forms and reports.

A. LOGICAL DATABASE DESIGN

The first part of the logical design process involves transforming the objects, as defined in chapter two and exhibited in Appendix D, into relations using Kroenke and Dolan's relational database model methodology. Kroenke and Dolan state,

The relational model is based on the concept that data is organized and stored in two-dimensional tables called relations. You can think of a relation as a file, and of each row in the relation as a record [Ref. 12].

An object is transformed into a relation by "stretching out" its properties horizontally. The properties become

attributes of the object. Then, object instances (what Kroenke and Dolan call "records") can be filled in underneath the relation's structure (See Figure 3.1). Each row is known as a tuple and each column represents a field (also called an attribute).

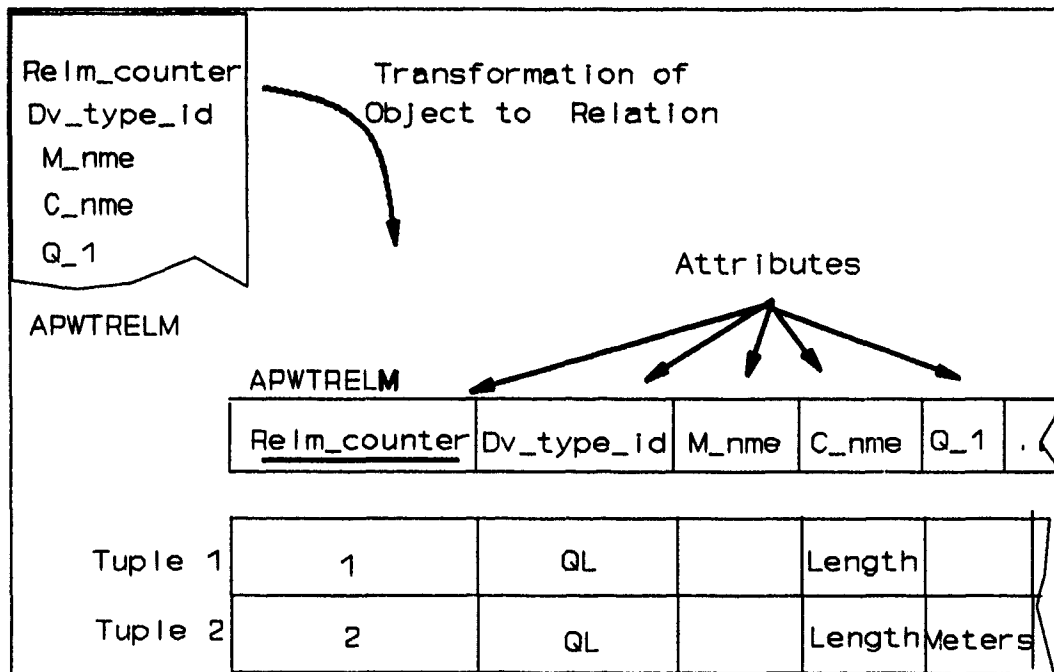


Figure 3.1. Object Transformation.

These attributes must be single-valued and have no repeating groups or arrays. Also, attributes must have a unique name, preventing creation of identical rows (or tuples) that would constitute duplication of data. A "key" is one or more fields or attributes, the value(s) of which uniquely identifies a row. Use of unique keys prevents data duplication. In Figure 3.1, each record has a unique

reference element number (Relm_counter) which serves as the key.

Each object becomes a set of related attributes or properties. The process of gathering properties into relations is known as normalization. Normalization is an important aspect of database design and will be applied to the design of the prototype dictionary. However, normalization "normal form" levels will not be discussed in detail.

In the second part of the logical design, binary relationships are used to link or join records in different relations, showing the dependencies between them. There are three types of basic binary relationships: one-to-one (1:1), one-to-many (1:N), and many-to-many (M:N). The binary relationships between relations can be used to build a tree structure that depicts how relations are linked and depend upon one another. This tree is called a relational diagram. The relational diagram, Figure 3.2, shows the prototype dictionary links. Lines or branches show the links. All 13 relations participate in one-to-many relationships as evidenced by a fork on one side of the branch. For example, a Class Word (APWTCWOR) may be in multiple Reference Elements (APWTRELM), but a Reference Element can only have one Class Word. Thus, constraints between relations are

identified in the diagram. Other constraints in these relationships are shown by placing symbols on the branches. The short horizontal line means "required" and the small circle connotes "optional." Returning to the example above, a Class Word may be optionally present in a Reference Element (it is independent), whereas, a Reference Element must have one and only one Class Word (it is functionally dependent on Class Word). Another symbol found within the relation box is an "F" which denotes a "foreign key." A foreign key is typically found in one-to-many relationships and occurs when the key of the parent is resident in that of the child. For example, APWTRELM has a 1:N relationship with APWTREFD (see Figure 3.2). Therefore, the Relm_counter attribute of APWTRELM is resident in the child, APWTREFD, as a foreign key.

The AR 25-9 naming conventions shown in Appendix C contain rules that involve relationships which are subsequently captured in the relational diagram. Appendices D and E provide object diagram structure and object property domain definitions respectively. Specific descriptions and relationships of each of the thirteen relations are shown in Appendix F.

The relational diagram shows the optional and required links and dependencies between objects and thus provides a

transition from the logical design to a physical design (transformation of objects into relations). Identification of relations and their relationships provides a foundation from which application design can take place.

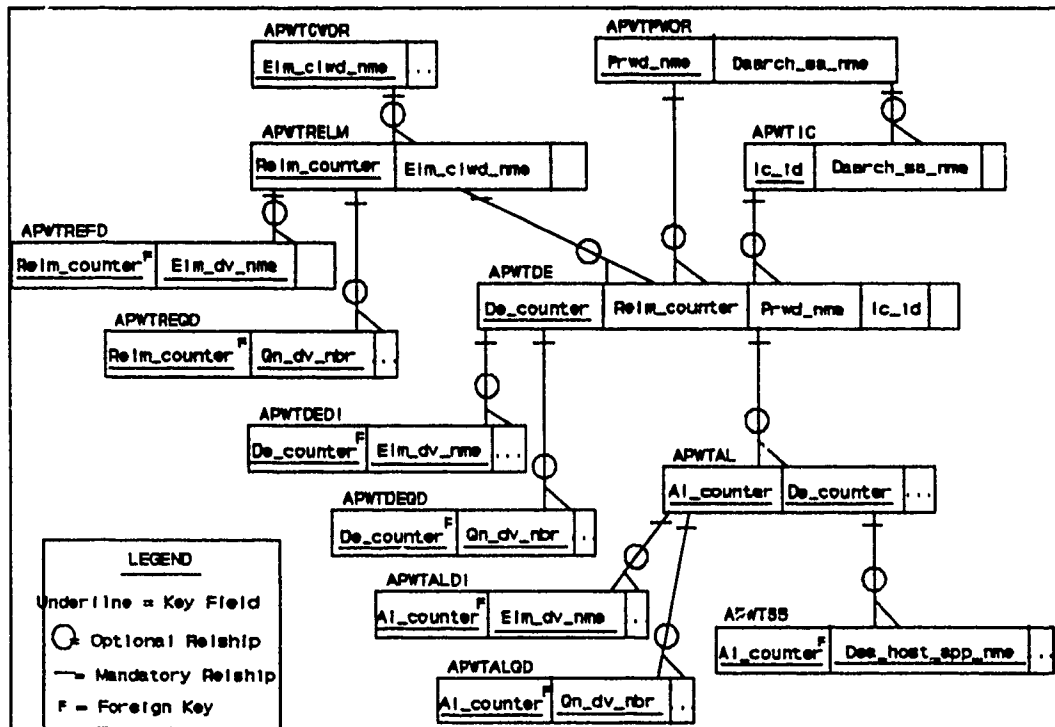


Figure 3.2. Prototype Dictionary Relational Diagram.

B. APPLICATION DESIGN

Application design is important because it includes the interface that the user sees, hears, and physically manipulates. The interface is the system to the user. An otherwise great system with a poor interface may not be embraced because it is perceived as too "user-hostile" or time consuming to use. Application design issues that must

be considered are: (1) Scope of the application functions;
(2) Menu hierarchy and menu, form and report
materializations [Ref. 13].

1. Application Functions Scope

The scope of the prototype dictionary application is centered around three standard functions:

1. User processing control functions;
2. Object query, print, and update functions;
3. Database security and integrity functions.

There are three basic types of application control mechanisms: command-oriented, menu-driven, or icon-driven. The control mechanism that the prototype dictionary uses is menu-driven. Menus permit easy selection of application functions as well as control access to application functions (a user can only select those options offered to him). Also, no commands need to be memorized which helps minimize training. A disadvantage of menus, however, is that they may become tedious to an expert or frequent user.

With respect to the second application function, objects by themselves provide incomplete information. It is the joining or querying of objects with links to other objects that yield the information that the user needs. For

example, if the prototype dictionary user desires to view all of the Data Items of a Reference Element, the relationship between Reference Element (APWTRELM) and Reference Element Data Value Item (APWTREFD) must be queried. The prototype dictionary supports such queries. Along the same lines, several different printed reports from similar queries can be generated (via menu selection) on all Reference Elements, Data Elements, Alias Elements, Class Words and Prime Words within this system. The update functions (add, edit and delete) are available to users as well.

The third function, database security and integrity is also a part of this system. Security is afforded through a password checking system upon initial logon into the system. Data integrity is maintained in different ways. First, masked forms are used in all of the update functions, grouping like attributes for better understanding and ensuring correct data input by permitting only the correct data type (character, integer, floating point, etc) to be entered in each of the fields. Second, some fields are "display-only." For example, if editing a record, the key field must be limited to "display-only" to prevent other previously linked relations from being linked incorrectly to a now newly created data record.

2. Menu Hierarchy and Materializations

Because the user desires a data-oriented system (dictionary), the menu hierarchy uses an object-to-action strategy. This means that choice of object, like Reference Element, is a first level choice, while actions, like add and delete, are second level choices. Figure 3.3 shows the

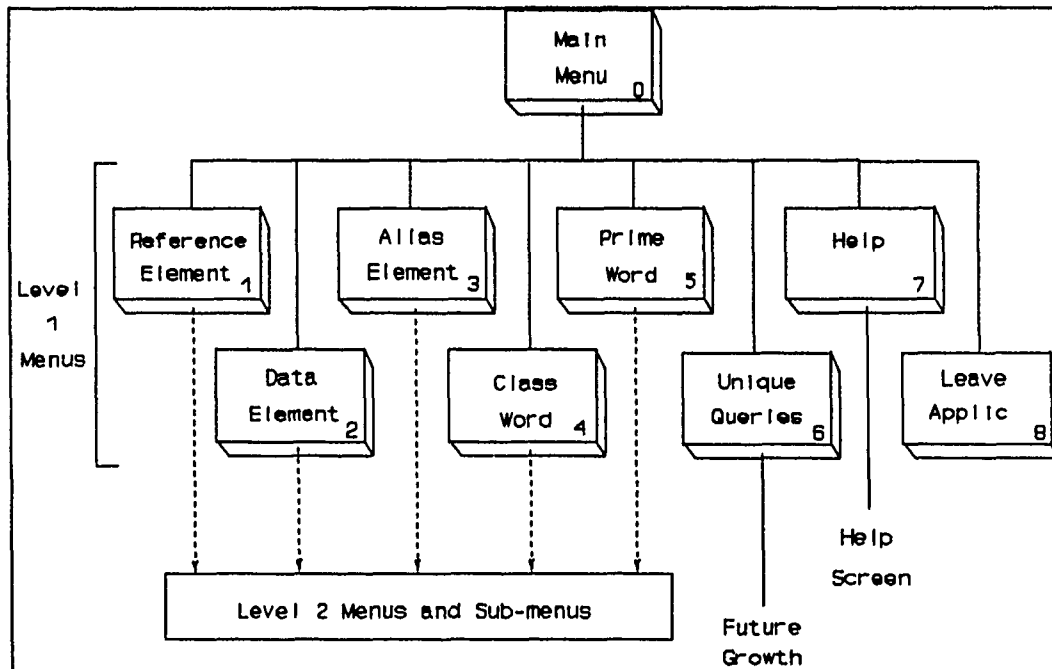


Figure 3.3. Main Menu Hierarchy.

main menu and menu hierarchy overview. Subsequent menu diagrams, though generic in nature, depict the functions of the application accurately.

a. Reference Element Option

Selection of the Reference Element option takes the user to the Reference Element menu where view, add,

edit, delete, print, menu help, and exit options are provided (see Figure 3.4). All form and report

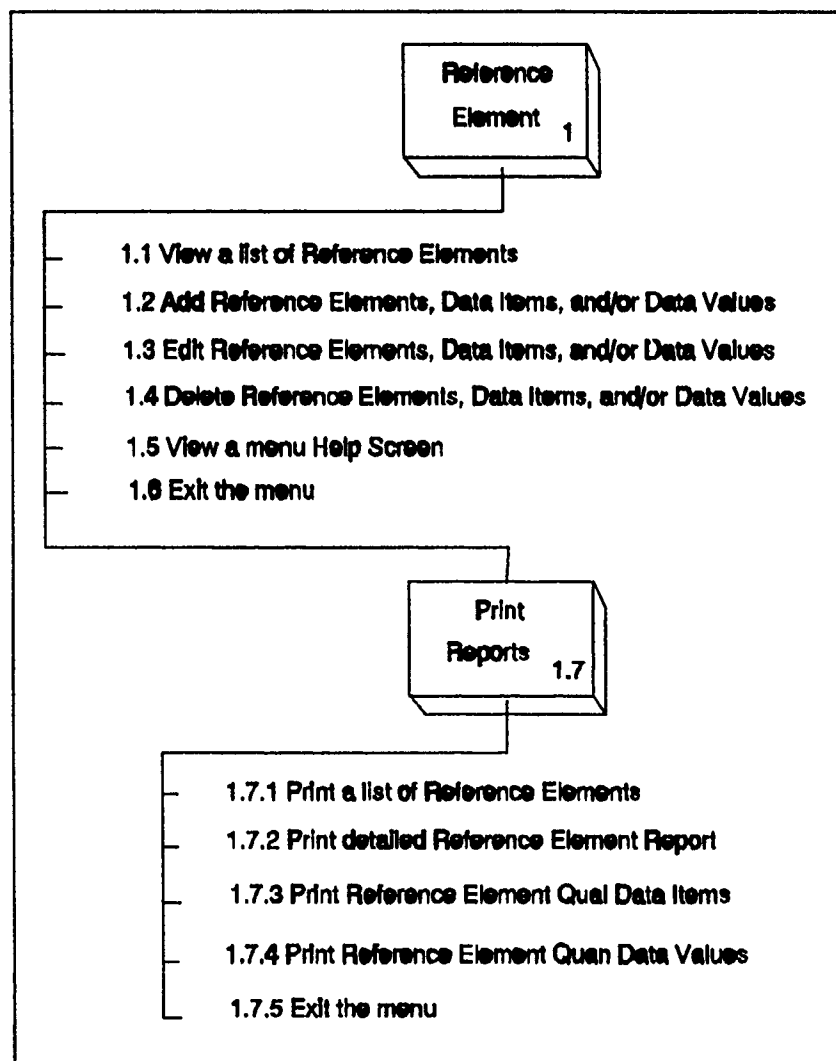


Figure 3.4. Reference Element Menu Options.

materializations are shown in Appendix G. The description of options follows:

(1) *Reference Element list view.* A one-page form is presented that shows Record number, Relm_counter number, Reference Element name and Data Value Type ID. The user may scroll through all of the values, but no updates are permitted.

(2) *Reference Element add.* A detailed four-page form is presented. The key field, Relm_counter, is required to continue the add sequence. Embedded within this form are Qualitative Data Item and Quantitative Data Value forms. This allows data items or data values associated with a Reference Element to be added at the same time. Also, it allows data items and data values to be added to existing Reference Elements.

(3) *Reference Element edit.* The user is first queried for a specific Relm_counter number to edit. From this input, a four-page form, similar to Reference Element add is presented. The key field, Relm_counter number, becomes display-only to preserve the integrity of other linked objects. All other fields may be edited. To prevent accidental deletion of a Reference Element, full deletion is not permitted within this function.

(4) *Reference Element delete.* The user is first queried for a specific Relm_counter number to delete. A three-page form is presented. In order to delete a

Reference Element, all associated Qualitative Data Items or Quantitative Data Values must first be deleted by the user. This removes the 1:N links and preserves the integrity of the database because these values may not exist without a parent Reference Element.

(5) *Help Screen.* This help screen defines what operations are available under each menu option.

(6) *Exit option.* This selection allows the user to return to the next higher menu in the hierarchy.

(7) *Print Report options.* This is a sub-menu for selection of the following printed report options:

1. Print a list of Reference Elements including Record number, Relm_counter number, Reference Element name, and Data Value Type ID.
2. From input of a valid Relm_counter number, print a detailed report on the Reference element using all of the attributes in the Reference Element object.
3. From input of a valid Relm_counter number, print the Qualitative Data Items and their definitions for a specific Reference Element.
4. From input of a valid Relm_counter number, print the Quantitative Data Values and their definitions for a specific Reference Element.
5. This selection enables the user to return to the next higher menu in the hierarchy.

b. Data Element Option

Selection of the Data Element option takes the user to the Data Element menu where view, add, edit, delete, print, menu help, and exit options are provided (see Figure 3.5). All form and report materializations are shown in Appendix G. The description of options follows:

(1) *Data Element list view.* A one-page form is presented that shows Record number, De_counter number, Data Element name and Data Value Type ID. The user may scroll through all of the values, but no updates are permitted.

(2) *Data Element add.* A detailed four-page form is presented. The key field, De_counter, is required to continue the add sequence. Embedded within this form are Qualitative Data Item and Quantitative Data Value forms. This allows data items or data values associated with a Data Element to be added at the same time. Also, it allows data items and data values to be added only to existing Data Elements.

(3) *Data Element edit.* The user is first queried for a specific De_counter number to edit. From this input, a four-page form, similar to add is presented. The key field, De_counter number, becomes display-only to preserve the integrity of other linked objects. All other

fields may be edited. To prevent accidental deletion of a Data Element, full deletion is not permitted within this function.

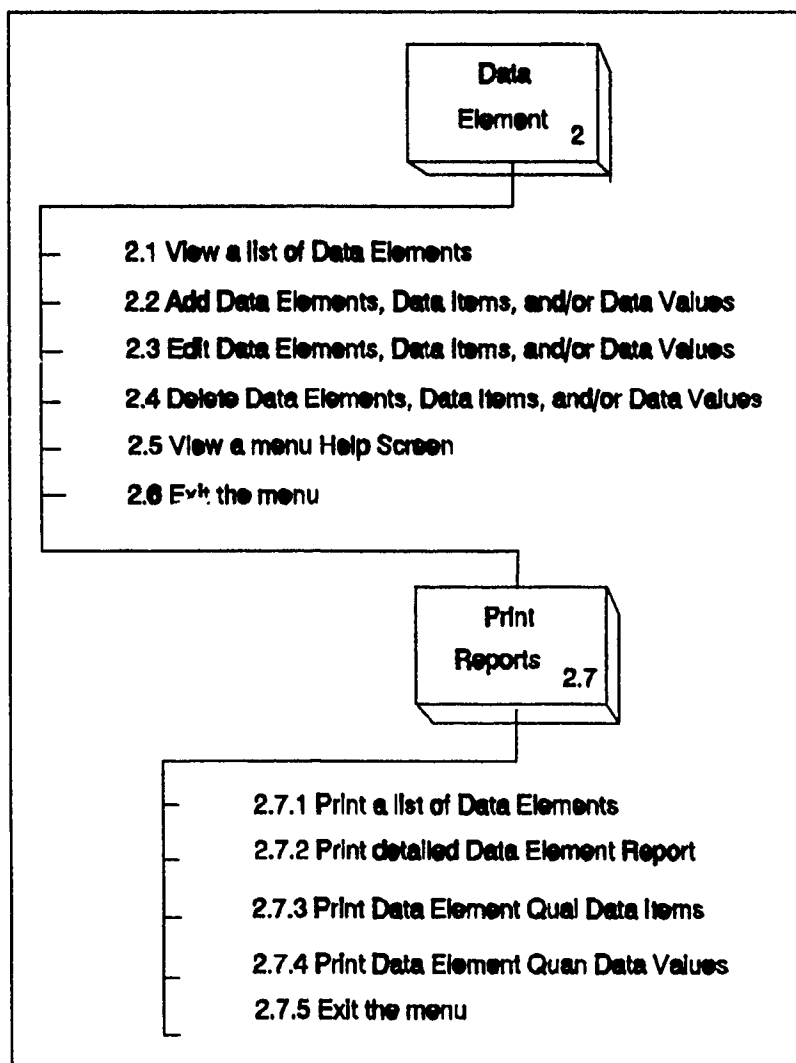


Figure 3.5. Data Element Menu Options.

(4) *Data Element delete.* The user is first queried for a specific De_counter number to delete. A three-page form is presented. In order to delete a Data

Element, all associated Qualitative Data Items or Quantitative Data Values must first be deleted. This removes the 1:N links and preserves the integrity of the database because these values may not exist without a parent Data Element.

(5) *Help Screen.* This help screen defines what operations are available under each menu option.

(6) *Exit option.* This selection allows the user to return to the next higher menu in the hierarchy.

(7) *Print Report options.* This is a sub-menu for selection of the following printed report options:

1. Print a list of Data Elements including Record number, De_counter number, Data Element name, and Data Value Type ID.
2. From input of a valid De_counter number, print a detailed report on the Data Element using all of the attributes in the Data Element object.
3. From input of a valid De_counter number, print the Qualitative Data Items and their definitions for a specific Data Element.
4. From input of a valid De_counter number, print the Quantitative Data Values and their definitions for a specific Data Element.
5. This selection enables the user to return to the next higher menu in the hierarchy.

c. Alias Element Option

Selection of the Alias Element option takes the user to the Alias Element menu where view, add, edit, delete, print, menu help, and exit options are provided (see Figure 3.6). All form and report materializations are shown in Appendix G. The description of options follows:

(1) **Alias Element list view.** A one-page form is presented that shows Record number, Al_counter number, Alias Name, and associated De_counter numbers. The user may scroll through all of the values, but no updates are permitted.

(2) **Alias Element add.** A detailed four-page form is presented. The key field, Al_counter, is required to continue the add sequence. Embedded within this form are Qualitative Data Item, Quantitative Data Value, and Host System Application Data forms. This allows data items or data values associated with an Alias Element to be added at the same time. Also, it allows data items, data values, and host system application data to be added only to existing Alias Elements.

(3) **Alias Element edit.** The user is first queried for a specific Al_counter number to edit. From this input, a four-page form, similar to add is presented. The key field, Al_counter number, becomes display-only to

preserve the integrity of other linked objects. All other fields may be edited. To prevent accidental deletion of an

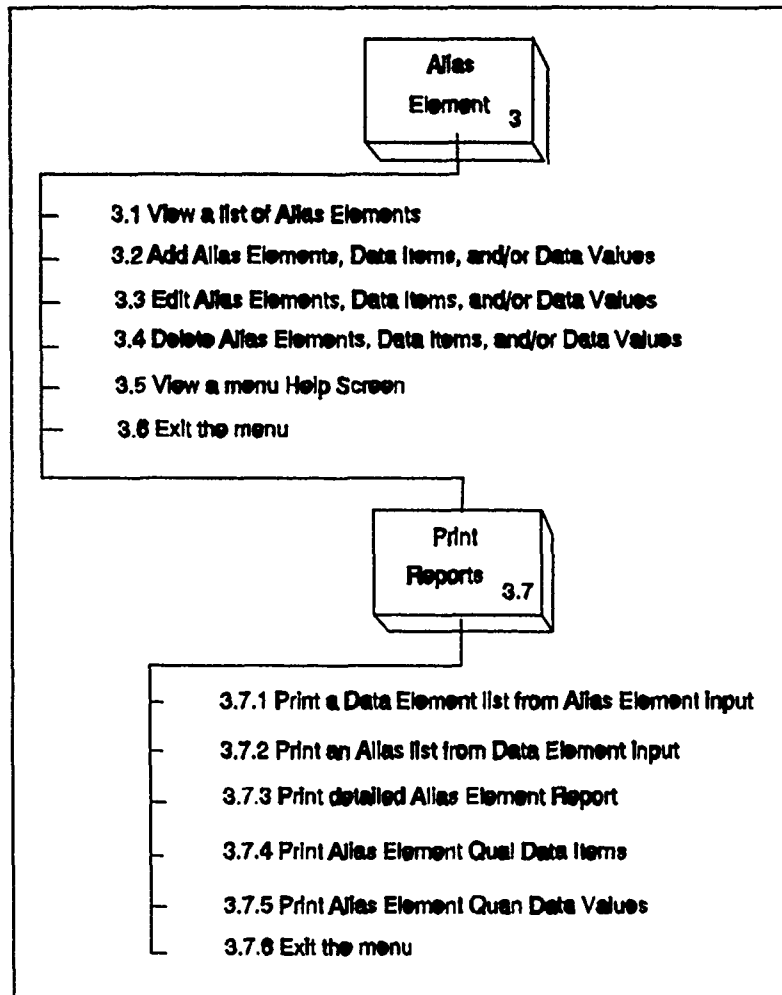


Figure 3.6. Alias Element Menu Options.

Alias Element, full deletion is not permitted within this function.

(4) *Alias Element delete.* The user is first queried for a specific Al_counter number to delete. A four-page form is presented. In order to delete an Alias

Element, all associated Qualitative Data Items, Quantitative Data Values, and Host System Application Data must first be deleted. This removes the 1:N links and preserves the integrity of the database because these values may not exist without a parent Alias Element.

(5) *Help Screen.* This help screen defines what operations are available under each menu option.

(6) *Exit option.* This selection allows the user to return to the next higher menu in the hierarchy.

(7) *Print Report options.* This is a sub-menu for selection of the following printed report options:

1. From input of a valid Al counter number, print a list of Data Elements that are associated with the Alias Element.
2. From input of a valid De counter number, print a list of Alias Elements that are associated with the Data Element.
3. From input of a valid Al counter number, print a detailed report on the Alias Element using all of the attributes in the Alias Element object and the Host System Application object.
4. From input of a valid Al counter number, print the Qualitative Data Items and their definitions for a specific Alias Element.
5. From input of a valid Al counter number, print the Quantitative Data Values and their definitions for a specific Alias Element.
6. This selection enables the user to return to the next higher menu in the hierarchy.

d. Class Word Option

Selection of the Class Word option takes the user to the Class Word menu where view, add, edit or delete, print, and exit options are provided (see Figure 3.7). All

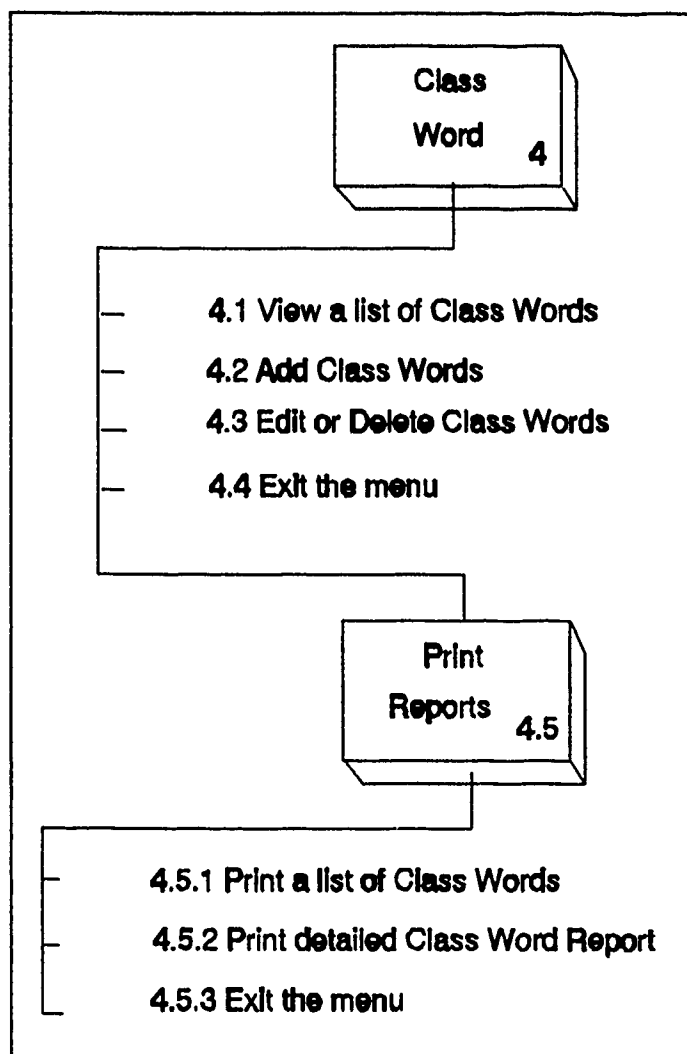


Figure 3.7. Class Word Menu Options.

form and report materializations are shown in Appendix G. the description of options follows:

(1) *Class Word list view.* A one-page form is presented that shows Record number and Class Word. The user may scroll through all of the values, but no updates are permitted.

(2) *Class Word add.* A detailed one-page form is presented. The key field, *Elm_Clwd_Nme* is required to continue the add sequence. This helps maintain data integrity.

(3) *Class Word edit or delete.* The user is first queried for a specific *Elm_Clwd_Nme* to edit or delete. These two functions, edit and delete were combined into one because there are only a few attributes for this object. Also, Class Words come from a predetermined and finite domain and it is anticipated that there will be little change of the records that exist.

(4) *Exit option.* This selection allows the user to return to the next higher menu in the hierarchy.

(5) *Print Report options.* This is a sub-menu for selection of the following printed report options:

1. Print a list of Class Words.
2. From input of a valid *Elm_Clwd_Nme*, print a detailed Class Word report.
3. This selection enables the user to return to the next higher menu in the hierarchy.

e. Prime Word Option

Selection of the Prime Word option takes the user to the Prime Word menu where view, add, edit or delete, print, and exit options are provided (see Figure 3.8). All form and report materializations are shown in Appendix G. The description of options follows:

(1) *Prime Word list view.* A one-page form is presented that shows Record number, Prime Word, and Army Data Subject-area. The user may scroll through all of the values, but no updates are permitted.

(2) *Prime Word add.* A detailed one-page form is presented. The key field, Prwd_Nme is required to continue the add sequence. This helps maintain data integrity.

(3) *Prime Word edit or delete.* The user is first queried for a specific Prwd_Nme to edit or delete. These two functions, edit and delete were combined into one because there are only a few attributes for this object. Also, Prime Words come from a predetermined and finite domain and it is anticipated that there will be little change of the records that exist.

(4) *Exit option.* This selection allows the user to return to the next higher menu in the hierarchy.

(5) *Print Report options.* Because this object only has two objects the only printed report is a list of Prime Words and their associated Army Data Subject-area.

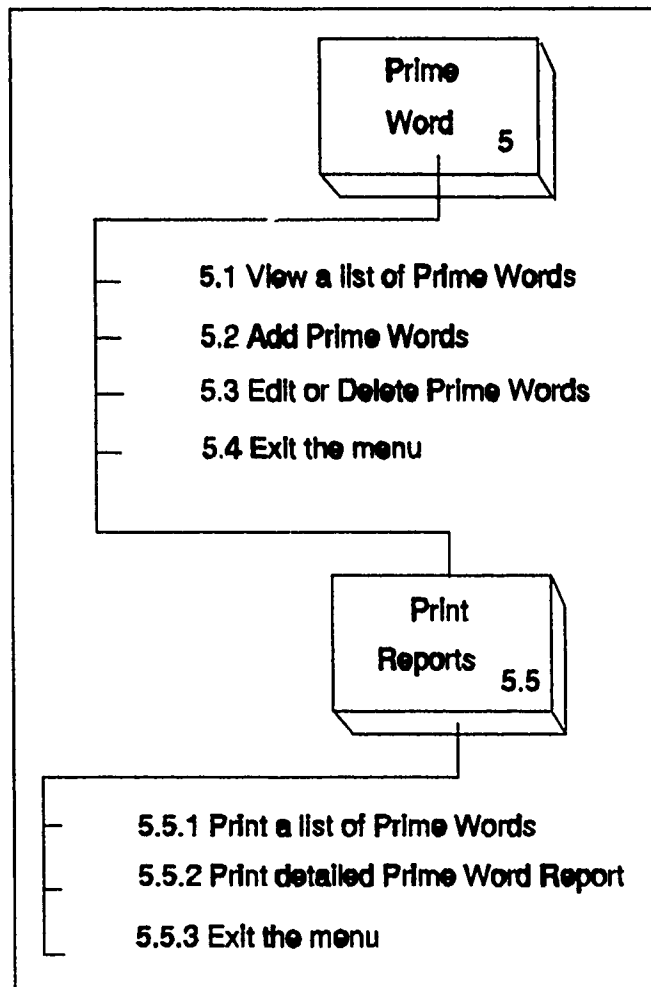


Figure 3.8. Prime Word Menu Options.

f. Unique Queries Option

This menu option is reserved for future use.

g. Help Option

Selection of the Help option moves the user to a one-page help screen. This screen shows a diagram of the menu hierarchy and provides some guidance on how to navigate through the menu hierarchy.

h. Leave Application Option

Selection of this the Leave option generates a "yes-no" question that the user must answer. "Yes" lets the user exit the application. "No" returns the user to the main menu.

3. Summary

This chapter has discussed Logical Database Design, the transformation of objects into relations and their relationships, and Application Design, the scope of functions of the prototype system and menu, form, and report materializations.

To this point, requirements definition, design, and application design phases have been largely generic. The next chapter will deal with implementation issues in connection with the host DBMS, Paradox 3.0.

IV. IMPLEMENTATION PHASE

The Definition, Design, and Application Design phases have been accomplished using "generic" methodologies, such as the Yourdon methodology (Data Flows) and the object-oriented/relational methodology. The implementation phase involves collecting the data descriptions, logical designs (objects and relations), functions and applications and then constructing the physical database and applications using a selected relational DBMS. The implementation phase addresses the choice of a relational DBMS (Paradox 3.0), the construction of the database using the relational DBMS, the problems and benefits found, and the user's manual and application code.

A. SELECTION OF PARADOX 3.0

Over the past few years, the microcomputer world has been dominated by Ashton-Tate's DBase III, III+, and IV products in the relational database arena. With technological improvements in microcomputer processing power (8088 to 80286 to 80386 cpu's and beyond), RAM capacity and speed, and secondary memory storage volume and speed, the demand for more powerful, flexible and user-friendly

relational databases has encouraged new entrants into the market place. An Infoworld study, involving the exhaustive comparison of six relational databases, rated Paradox 3.0 the highest with a score of 9.5 out of a possible 10.0 [Ref. 14]. Benchmarks included the categories of performance, documentation, ease of learning, ease of use, error handling, and support (See Figure 4.1).

Paradox was found to be "the best all around relational database for interactive and development use." Other major systems, such as ORACLE and Ingres were considered, however the most current versions of these products exceeded the RAM limitations listed in Chapter I. Paradox falls within all of the hardware constraints listed in Chapter I.

Paradox 3.0 proved to be a very flexible and capable relational DBMS for building this prototype.

B. CONSTRUCTION OF THE PROTOTYPE USING PARADOX 3.0

Paradox 3.0 offers three different methods for construction of a database system: the standard design module, the Personal Programmer (Application Generator), and the Paradox Application Language (PAL) editor, Paradox's structured programming language. [Ref. 15]. Program capacities include:

INFOWORLD COMPARISON		Clarion Prof. (WT.)	DBase IV 1.0	Foxpro 1.0	Informix- SQL 2.10.08	Pendax 3.0	Ribase for DOS 2.11
Performance							
Rel. Data Entry	(75)	E	P	E	VG	E	E
Rel. Querying	(75)	G	P	G	E	E	E
Rel. Reporting	(100)	E	VG	VG	VG	E	E
Programing Lang.	(100)	E	VG	VG	E	E	E
Speed of Rel. Operations	(125)	E	G	E	S	VG	S
Documentation	(100)	VG	S	E	E	E	E
Ease of Learning	(50)	VG	P	G	VG	E	E
Ease of use	(125)	E	P	VG	G	E	E
Error Handling	(100)	E	U	S	VG	E	VG
Support							
Support Policies	(25)	G	G	VG	S	VG	S
Technical Support	(50)	G	P	S	VG	VG	VG
Value	(75)	E	P	VG	VG	E	E
Final Scores		9.0	4.0	7.7	7.6	9.5	9.9
E = Excellent = 1.0 VG = Very Good = 0.75 G = Good = .625		S = Satisfactory = 0.5 P = Poor = 0.25 U = Unacceptable or N/A = 0.0					

Figure 4.1. Infoworld Study Results.

1. 2 billion records with 4000 characters each.
2. 255 fields with up to 255 characters per field.
3. Each table may have up to 15 associated forms and 15 reports.
4. Unlimited number of tables that may be joined in a query.
5. The Personal Programmer allows 15 tables, 15 selections per menu, and 10 levels of menus in a given application.

Development of the prototype involved a combination of all three methods. Fourteen tables were created from the original objects using the standard module. Generation of fourteen instead of the logical design of thirteen occurred because the Data Element table, representing the Data Element object (APWTDE), exceeded record capacity for a "keyed" table, 1350 bytes. Therefore, APWTDE had to be divided into two parts, APWTDE and APWTDE2, and then linked relationally in order to satisfy memory requirements. All forms, reports and queries were created in the standard module (See Appendix G).

One of the strengths of Paradox is its form generator. A form, based on a single table, may have other forms from other tables "embedded" within it. Maximum form length in Paradox 3.0 is four pages, allowing up to four embedded forms. The embedded form concept is very helpful in

maintaining the integrity of the data within the database. In the prototype, the embedded form allowed the Qualitative Data Items and Quantitative Data Values of Reference, Data and Alias Elements to be added, edited and deleted in connection with all of the regular descriptive attributes associated with those elements. This capability helped to preserve data integrity because Values and Items could not exist without their parent elements. Without embedded forms, deletion of an element would first involve going to a separate Values or Items form, performing deletion, and then returning to the associated element and perform a deletion on it. Another strength of the form generator consisted of the ability to generate "validity checks." These validity checks enable fields to be constrained in the following ways:

1. Required Field Data Entry. Required entry exists for key fields in Reference, Data, and Alias Element tables as well as Prime and Class Word tables.
2. Field Data Format. A field can be constrained to accept only numbers or alphanumerics or variations of the two, including capitalization and use of characters.
3. Look-up Tables. Look-up tables are used to limit the domain of the content of a field. For example, Reference Elements are built from a finite list of Class Words. Thus, by pressing [F1], a look-up table of Class Words is presented to the user for selection. This function greatly preserves data integrity and

reinforces the Element naming conventions discussed in Chapter II.

4. Default Values. Default values are placed in some fields in each form. For example, if the Element Status ID field is bypassed, the field automatically shows "PR" for Proposed Element.

The forms generator was easy to use and flexible. Calculated Field functions were used to add the various modifiers, qualifiers, Prime Words and Class Words together to form Reference and Data Elements. Also, form color options and ASCII character options were available to make the forms pleasing to the eye as well as highlight important sections of the form.

Like the form generator, the report generator was available within the standard module. The report generator permitted two styles: Columnar and free form report. Both styles were used. All "list" reports were generated by the columnar option; while, the free form report style was used for the "detailed reports."

The report generator was not as user-friendly. For example, a detailed report for a Data Element does not include Qualitative Data Items or Quantitative Data Values. Therefore the Items and Values of a Data Element are reported separately. Also, report generator operations were somewhat slower than the form generator.

Queries were constructed in both the standard module and the Personal Programmer. In the standard module, Query-By-Example (QBE) methodology is used to relate tables. QBE is a simple, yet capable methodology that presents the table structure to the developer and then lets the developer link common fields and place constraints on field values in several different tables in order to derive desired information. QBE was used to link Data Values and Items to their parent elements (Data, Reference, and Alias). In the Personal Programmer, query options were more detailed. For example, in the application, if the user desires to edit a Data Element, he is first queried for the Data Element Number (De_Counter). Queries are used in almost every menu option to narrow-down and then quickly comply with the user needs. The Personal Programmer is the main applications generator which brings all of the forms, reports and queries together.

The Personal Programmer performs like a "General Contractor." It takes the parts, subcontractors, and organizes them into a structure that is cohesive and coordinated. Using a menu hierarchy structure, the Personal Programmer generates "scripts" (applications) that run much like a giant batch file. Scripts are really recorded keystrokes. However, scripts are also the "gateway" to PAL

since they are recorded in PAL and can be edited and enhanced. Scripts are interpreted by Paradox and then executed. Though slower than compiled code, Paradox 3.0 is fast when compared to similar products in the market place [Ref. 12].

A single master script was generated for the prototype. However, the script was so large that there was insufficient RAM to run it. Because of this, the master script was divided into smaller scripts which were chained together. The major script hierarchy and definition is shown in Figure 4.2. Smaller scripts are also used to perform utility functions. The advantage of scripts is that they break the prototype into more manageable modules. When an update is required, only the specific module needs to be changed.

TSR Corporation, located in New York, is currently developing a compiler for Paradox 3.0. However, a "Runtime" program is available for extra cost, through Borland Corporation, which does not increase speed, but enables applications to be run without main program support, reducing the requirement of reserving approximately five megabytes of secondary memory for Paradox 3.0.

PAL, mentioned earlier, was used to enhance the scripts. The PAL editor was used to both add code as well as subtract

unnneeded code. Also, PAL helped document each of the scripts which was beneficial in the development process.

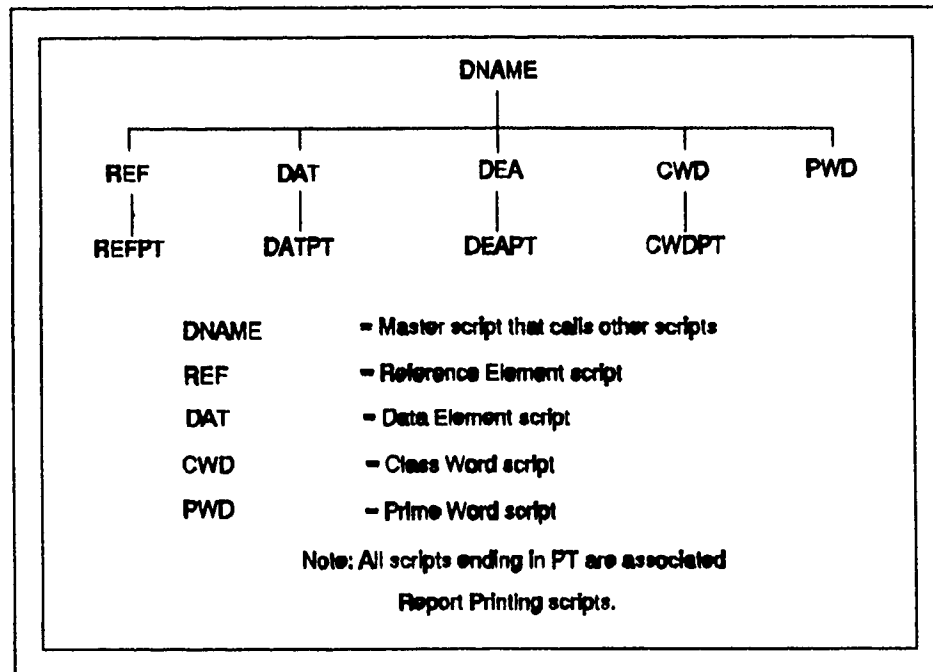


Figure 4.2. Prototype Script Hierarchy.

C. PROBLEMS AND BENEFITS OF PARADOX 3.0 IN IMPLEMENTATION

Although Paradox 3.0 proved to be a very capable relational DBMS, some problems were encountered. Problems and benefits are summarized as follows:

1. Problem. The keyed file memory limit of 1350 bytes caused a major table to be divided into two separate tables.
2. Problem. Selection of a Reference, Data, or Alias Element for deletion must be done by paging through the elements. The Personal Programmer will not permit selection by simple query in this case.

3. Problem. Where duplicate key field instances occur, both "matched" records are placed into a "Keyviol" (Key Violation) table. Reconciliation of this duplication is a cumbersome process.
4. Problem. Because Paradox offers many flexible options in building a database system, much RAM is used (up to 640K). The method of using smaller scripts, for this reason, was not identified in the documentation.
5. Problem. Speed. Though fast compared to other relational DBMS's, Paradox 3.0 is still considered slow compared to a compiled counterpart. For example, on a 12.5 MHZ 80286 IBM Clone microcomputer, a query to edit 1 of 50 Data Element records took an average of 32 seconds to display the four-page record (with three embedded forms).
6. Problem. Mouse Support. Though not a requirement for the prototype, the use of a mouse in the development stage would have accelerated the process. Long hours in the development process led to increased keyboard-entry errors. The mouse option would have alleviated the problem somewhat.
7. Benefit. Three separate methods (Standard Module, Personal Programmer and PAL Editor) for system development.
8. Benefit. Capable and flexible form generator, providing: embedded forms, validity checks and highlighting capability.
9. Benefit. Capable report generator.
10. Benefit. On-line help module for standard module and personal Programmer provided immediate answers to questions encountered during development.
11. Benefit. QBE methodology, used to generate queries and link relations proved to be simple and effective.
12. Benefit. Paradox 3.0 is a user-friendly and well-documented relational DBMS found to be superior in a comparison of six relational DBMS's [Ref. 14].

Overall, Paradox 3.0 benefits far outweighed the problems encountered in the development process.

D. USERS MANUAL AND SCRIPT CODE

The objective of the User's Manual is to provide a description of prototype operations at each step along the menu hierarchy. Although each screen contains basic instructions on what keys to use for certain functions, the main purpose of the manual is to provide amplification of instruction where questions on operations may arise. The manual is located in Appendix H. The code for the primary scripts shown in Figure 4.2 is located in Appendix I.

The Conclusions from development of the prototype and recommendations are provided in the following chapter.

V. CONCLUSION AND RECOMMENDATIONS

This thesis has discussed the importance of data element standardization as a foundation for standardizing Management Information Systems (MIS) used by all of the services within DOD. Data is a valuable resource that must be easily and effectively shared between the services. The interoperability of multiple MIS systems based on this standardization will create a reduction in both data and system redundancy, and result in an eventual cost savings. It will also create an increase in information sharing and more effective communication, providing a better return on dollars invested in MIS. To assist the standardization process, this thesis has identified the data requirements, the functional component requirements, the logical database design and the application design for a prototype microcomputer DBMS dictionary for standardizing, updating, viewing and storing data elements. This thesis then successfully implemented the system design and developed a prototype using a powerful relational DBMS, Paradox 3.0.

It is recommended that this prototype dictionary be evaluated by users in CIM and ODISC4 for verification and validation. Further, it is recommended that changes be

submitted to the Naval Postgraduate School Computer Technology department as a follow-on thesis topic. The "Unique Query" option, in the prototype's main menu, has been reserved for the purpose of updating user query needs. For example, the user may desire a listing of Data Elements based on Element Status (which Data Elements have been proposed, approved, etc.). Other thesis topics include:

1. Exploration of a data download capability, either by floppy disk or dial-up line, from the ADD ADSS to the prototype dictionary.
2. Design and development of an expert shell, used in conjunction with the prototype, that would enable the user to select a level of operation based on his computer skills (new user, experienced user, expert user).
3. Comparison of the prototype dictionary with any other DBMS systems that are created to help developers standardize and store Data Elements.

Because of DOD's new policy of standardizing MIS, it is important that thesis research in this area continue in order to reinforce the importance of the Data Element standardization goal.

APPENDIX A

A. PRIME WORD LIST

Rec Nbr	Prime Word Name	Army Data Subject-Area Name
1	Accounting	Budget
2	Acquisition	Acquisition
3	Administration	Support Activities
4	Affair	Public Affairs
5	Agency	
6	Agreement	Contracts
7	Air	Transportation
8	Air-Defence	Operations Plans
9	Air-Ground	Operations Plans
10	Aircraft	
11	Airfield	
12	Airlift	
13	Airport	
14	Alert	
15	Ammunition	
16	Anchorage	
17	Annex	Facilities
18	Appropriated	Funds
19	Apron	
20	Arctic	
21	Army	Unit(s) & Org(s)
22	Arresting-Gear	
23	Arrival	
24	Assessment	Operations Plans
25	Asset	
26	Assistance	Security Assistance
27	Audio	Information Management
28	Audit	Support Activities
29	Authorization	Structure
30	Automation	Information Management
31	Barrier	Operations Plans
32	Base	Facilities
33	Battlefield	Operations Plans

34	Bed	
35	Berth	
36	Biological	Operations Plans
37	Budget	Budget
38	Bunker	
39	Camp	Facilities
40	Capability	Operations Plans
41	Cargo	
42	Carrier	
43	Catastrophic	Crisis Operations
44	Cemetery	Facilities
45	Channel	
46	Chart	
47	Chemical	Operations Plans
48	Civil	
49	Civilian	Personnel
50	Clinic	Support Activities
51	Combat	
52	Command	Guidance and Doctrine
53	Commercial	Commercial Activities
54	Communication	Information Management
55	Community	Support Activities
56	Compensation	Funds
57	Complex	
58	Component	
59	Computer	Information Management
60	Congressional	Government Liaison
61	Construction	Facilities
62	Container	
63	Contract	Contracts
64	Contractor	Commercial Activities
65	Conversion	
66	Convoy	
67	Counterintelligence	Intelligence
68	Country	
69	County	
70	Craft	
71	Crane	
72	Crisis	Crisis Operations
73	Deception	Intelligence
74	Defense	Operations Plans
75	Departure	
76	Dependent	Personnel
77	Deployment	Operations Plans
78	Description	
79	Developer	Materiel
80	Development	Research and Development
81	Direction	Guidance and Doctrine

82	Directive	Guidance and Doctrine
83	Disaster	Crisis Operations
84	Disbursement	Funds
85	Discipline	Security
86	Diseased	
87	Document	
88	Documentation	Units(s) & Org(s)
89	Electricity	
90	Electronic	Research and Development
91	Encyclopedia	
92	Enemy	Intelligence
93	Engineering	Facilities
94	Equal-Opportunity	Personnel
95	Equipment	Materiel
96	Evacuee	
97	Evaluation	Operational Testing
98	Executive	Government Liaison
99	Exercise	Operations Plans
100	Experiment	Research and Development
101	External	Government Liaison
102	Facility	Facilities
103	Family	Personnel
104	Finance	Budget
105	Fire-Support	Operations Plans
106	Force	Readiness
107	Foreignfuel	Govre
108	Funds	Funds
109	Geographic	Intelligence
110	Goal	Unit(s) & Org(s)
111	Government	Government Liaison
112	Harbor	
113	Health	Support Activities
114	Hospital	Support Activities
115	Hostilities	
116	Housing	Facilities
117	Ice	
118	Individual	Personnel
119	Industrial	Acquisition
120	Information	Information Mangaement
121	Inspection	Support Activities
122	Installation	Support Activities
123	Institutional	Training
124	Intelligence	Intelligence
125	Interheadquarter	Government Liaison
126	International	Government Liaison
127	Interservice	Government Liaison
128	Intertheater	Operations Plans
129	Intratheater	Operations Plans

130	Inventory	Materiel
131	Investigation	Support Activities
132	Item	
133	Laboratory	Research and Development
134	Land	Transportation
135	Language	
136	Law-and-Order	Security
137	Legal	Support Activities
138	Liaison	Government Liaison
139	Library	Information Management
140	Life-Science	Research and Development
141	Lighter	
142	Local	Personnel
143	Location	
144	Logistic	Materiel
145	Long-Range	Operations Plans
146	Maintenance	Facilities
147	Major-Item	Materiel
148	Management	Unit(s) & Org(s)
149	Maneuver	Operations Plans
150	Manpower	Structure
151	Mapping	Intelligence
152	Materiel	Materiel
153	Medical	
154	Member	Personnel
155	Message	
156	Military	Personnel
157	Mission	
158	Mobilization	Operations Plans
159	Movement	Operations Plans
160	Nation	
161	National	Government Liaison
162	National-Guard	Unit(s) & Org(s)
163	Non-Appropriated	Funds
164	Non-Evacuee	
165	Nuclear	Operations Plans
166	Nuclear-Surety	Intelligence
167	Obstacle	Operations Plans
168	Offense	Operations Plans
169	Office	Facilities
170	Operation	Operations Plans
171	Operational	Operations Plans
172	Organization	Unit(s) & Org(s)
173	Outpatient	
174	POM	Army Program
175	Passenger	
176	Patient	
177	Personnel	Personnel

178	Petroleum	
179	Physical	Security
180	Pipeline	
181	Plan	Operations Plans
182	Policy	Guidance and Doctrine
183	Port	
184	Post	Facilities
185	Printing	Information Management
186	Priority	Guidance and Doctrine
187	Prisoner	Security
188	Procedure	Unit(s) & Org(s)
189	Production	Facilities
190	Program	Army Programs
191	Project	
192	Protocol	Research and Development
193	Psychological	Operations Plans
194	Public	Public Affairs
195	Publication	Information Management
196	Rail	Transportation
197	Railroad	
198	Ramp	
199	Range	Facilities
200	Readiness	Readiness
201	Receipt	Funds
202	Reconnaissance	Intelligence
203	Record	Information Management
204	Regulation	Guidance and Doctrine
205	Relief	Crisis Operations
206	Religious	Support Activities
207	Report	Unit(s) & Org(s)
208	Requirement	
209	Research	Research and Development
210	Reservation	Facilities
211	Reserve	Unit(s) & Org(s)
212	Resource	Army Program
213	Road	
214	Runway	
215	Safety	Support Activities
216	Sample	Research and Development
217	Science	Research and Development
218	Sea	Transportation
219	Seaport	
220	Security	Security
221	Security-Assistance	Security-Assistance
222	Sequence	
223	Service	
224	Ship	
225	Soldier	Support Activities

226	Special	Crisis Operations
227	Standard	Unit(s) & Org(s)
228	State	
229	Stock	
230	Storage	Facilities
231	Strategic	Guidance and Doctrine
232	Strategy	Guidance and Doctrine
233	Structure	Structure
234	Study	Studies Program
235	Subject	Research and Development
236	Supplier	Commercial Activities
237	Supply	Materiel
238	Support	Support Activities
239	Surveillance	Security
240	Tactical	Unit(s) & Org(s)
241	Technology	Research and Development
242	Telecommunications	Information Management
243	Terminal	Facilities
244	Test	Operational Testing
245	Theater	
246	Tidal	
247	Topology	Intelligence
248	Training	Training
249	Transport	Operations Plans
250	Transportation	Transportation
251	Tugboat	
252	Unconventional	Crisis Operations
253	Union	Personnel
254	Unit	Unit(s) and
Organization(s)		
255	Vehicle	
256	Vendor	Commercial Activities
257	Visual	Information Management
258	War	
259	Warfare	Operations Plans
260	Water	
261	Weather	Intelligence
262	Wharf	
263	Work	Civil Works
264	Zone	

APPENDIX B

A. CLASS WORD LIST

Record Number	Class Word Name
1	Acceleration
2	Amount
3	Angle
4	Area
5	Category
6	Code
7	Cost
8	Date
9	Date-Time-Group
10	Density
11	Depth
12	Distance
13	Flow
14	Height
15	Humidity
16	Identifier
17	Latitude
18	Length
19	Location
20	Longitude
21	Mass
22	Name
23	Number
24	Power
25	Pressure
26	Quantity
27	Range
28	Size
29	Temperature
30	Tension
31	Text
32	Time
33	Torque
34	Velocity

35	Viscosity
36	Volume
37	Weight
38	Width
39	Year

APPENDIX C

A. AR 25-9 Naming Conventions

1. Rule 1

A reference element name will contain one and only one class word. Comment: In this way, the standard element is formulated to describe only one type of information collected about an object.

2. Rule 2

Class words will be reserved; that is, they will not be used as modifiers, qualifiers or prime words.

3. Rule 3

Each data element will contain one designated prime word. and describe only one concept. Comment: Having only one prime word, the data element is explicitly formulated to describe only one concept.

4. Rule 4

The sequence of words in a data element will be as follows: modifier(s) (if required), prime words, modifier(s) (if required), class word, qualifier(s) (if required).

5. Rule 5

Each data element name will include its reference element name.

6. Rule 6

Plurals of class words or prime words are not permitted. Comment: Removing plurals from data element names encourages the designer to think in terms of primitive concepts and increases the possibility that two people will develop the same name to describe identical concepts.

7. Rule 7

Modifiers and qualifiers will be used to describe a standard element fully. Six modifiers per prime word and one modifier with two qualifiers per class word are allowed.

8. Rule 8

The word order of commonly used terms will be preserved in data element alias names (for example Port of debarkation and Department of Defense).

9. Rule 9

A unit of measure suffix may be added to the names of all elements that describe a numeric quantity (for example Volume-in-Liters).

10. Rule 10

No abbreviations or acronyms are permitted in the standard element name. Comment: Abbreviations and acronyms would reduce the clarity of the standard element name.

11. Rule 11

Only alphabetic characters (A-Z) are permitted in standard element names. Comment: There are two exceptions to Rule 11: (1) a hyphen may be used to connect the words in a prime term or reference element name; (2) a number may be used when it is part of a descriptive name (for example, M109A3 Howitzer). Comment: Permitting only alphabetic characters encourages standard element developers to name standard elements in terms of what the data are and not how they are stored or used. This rule also increases the probability that different people will develop the same name for identical standard elements.

12. Rule 12

Names of organizations, computer or information systems, directives, forms, screens, or reports are not permitted in standard element names.

13. Rule 13

Titles of blocks, rows, or columns of screens, reports, or listings are not permitted in standard element names unless those titles satisfy rules 1-11.

APPENDIX D

Relm_counter
 Dv_type_id
 M_nme
 C_nme
 Q_1
 Q_2
 Relm_ver_nbr
 Relm_nme
 Relm_creator_id
 Elm_status_id
 Elm_appvl_dto
 Elm_mod_dto
 Area_rvw_st
 Elm_dtypo_eat
 Elm_std_auth_id
 Elm_adec_nme
 Elm_max_dv_lgthchr
 Elm_just_eat
 Elm_dv_list_txt
 Elm_def_txt
 Elm_dom_def_txt
 Qn_lrng_nbr
 Qn_hrng_nbr
 Qn_scale_nbr
 Elm_cmt_txt

APWTREQD

MV

APWTREFD

MV

APWTDE

MV

APWTCWOR

Qs_dv_nbr

Elm_dv_def_txt

APWTRELM

APWTREQD

Elm_dv_nme

Elm_dv_def_txt

APWTRELM

APWTREFD

Elm_clwd_nme

Elm_def_txt

lc_eat_nme

Dv_type_id

APWTRELM

MV

APWTCWOR

NOTES:

APWTREQD - Ref Elmnt Data
Value Number

APWTREFD - Ref Elmnt Data Item

APWTDE - Data Element

APWTCWOR - Class Word

APWTRELM

De_Counter
 Dv_Type_ID
 De_Nme
 Modifier_1
 ↓
 Modifier_12
 De_Ver_Nbr
 De_Creator_ID
 Elm_Status_ID
 Elm_Appvl_Dte
 Elm_Mod_Dte
 Area_Rvw_ST
 De_Mnmnic_Abb
 Elm_Adoc_Nme
 De_Resp_Ofc_Nme
 Ic_ID
 De_Secu_Cat
 Elm_Std_Auth_ID
 De_Timness_ID
 Elm_Max_Dv_Lgt hchr
 Ql_Dv_Acry_Nbr pcl
 Qn_Lrng_nbr
 Qn_Hrng_nbr

Qn_Scale_Nbr
 Qn_Acry_ID
 Elm_Def_Txt
 Elm_Dom_Def_Txt
 Elm_Dv_List_Txt
 Elm_Cmt_Txt
 De_Calc_Fmla_Txt

APWTIC

APWTRELM

APWTAL

MV

APWTDEDI

MV

APWTDEQD

MV

APWTPWOR

APWTDE

Note:

APWTRELM = Ref Elmnt

APWTAL = Alias

APWTDEDI = Data Elmnt Data Item

APWTDEQD = Data Elmnt Data Value

APWTPWOR = Prime Word

El m_Dv_Nme
El m_Dv_Def_Txt
APWTDE

APWTDEDI

Pr wd_Nme	
Da arch_Sa_Nme	
APWTIC	MV
APWTDE	MV

APWTPWOR

Qn_Dv_Nbr
El m_Dv_Def_Txt
APWTDE

APWTDEQD

Note:

APWTIC - Info Class

Ic_Id	
Ic_Nme	
Da arch_Sa_Nme	
In_Proc_Nme	
In_Propnt_Nme	
APWTPWOR	
APWTDE	MV

APWTIC

Al_Counter
 Dea_Nme
 Elm_Creator_ID
 Elm_Just_Cat
 Elm_Dv_List_Txt
 Elm_Max_Dv_Lgt hehr
 Elm_Dom_Def_Txt
 Ql_Dv_Acry_Nbrpet
 Qn_Lrng_Nbr
 Qn_Hrng_Nbr
 Qn_Scale_Nbr
 Qn_Acry_ID
 DE_Calc_Fmla_Txt
 Elm_Cmt_Txt
 Dv_Type_ID

APWTALDI

MV

APWTALQD

MV

APWTSS

MV

APWTDE

APWTAL

Elm_Dv_Nme

Elm_Dv_Def_Txt

APWTAL

APWTALDI

Qn_Dv_Nbr

Elm_Dv_Def_Txt

APWTAL

APWTALQD

Dea_Host_App_Nme

Dea_Host_Sys_Nme

Dea_Int_Fmt_Cat

Dea_Resp_Ofc_Nme

APWTAL

APWTSS

Note:

APWTALDI - Alias Data Val Name

APWTALQD - Alias Data Value Number

APWTSS - Alias System

APWTAL - Alias

APPENDIX E

A. DOMAIN DEFINITION AND DESCRIPTION OF OBJECT PROPERTIES

Object properties are actual Data Elements within the prototype dictionary.

Record Number: 1

```
-----  
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth  
-----  
          1      1      Relm-Nme          QL          80
```

Data Element Name

```
-----  
Information Reference Element      Name
```

Element Definition Text

```
-----  
A character string given to a reference element based on  
a class word that identifies a domain. (See comment text).
```

Domain Definition Text

```
-----  
A generic domain comprised of the following ASCII  
characters: A - Z; Hyphen (-); and Underscore (_).
```

Record Number: 2

```
-----
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth
-----  -
          2      1      Dv-Type-ID        QL          2
```

Data Element Name

Information	Data Value Type	Identifier
-------------	-----------------	------------

Element Definition Text

An indication of the data value type of an information element.

Domain Definition Text

A specific domain comprised of the following qualitative data values: QN = Quantitative Data; QL = Qualitative Data.

Record Number: 3

```
-----
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth
-----  -
          3      1      Elm-Clwd-Nme      QL          20
```

Data Element Name

Information Element Class Word Name

Element Definition Text

A character string from a reserved word list that identifies the reference element domain.

Domain Definition Text

A specific domain comprised of the qualitative data values listed in appendix I of AR 25-9.

Record Number: 4

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Length
4	1	Elm-Mod_nme-xx	QL	27

Data Element Name

Information Element Modifier Name

Element Definition Text

A character string that further describes a characteristic of an object, a relationship between

objects, or the object itself.

Domain Definition Text

A generic domain comprised of the following ASCII
characters: A-Z, Hyphen (-); and Underscore (_).

Record Number: 5

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Lngh
5	1	Relm-Qlf-Nme-xx	QL	15

Data Element Name

Information Reference Element Qualifier	Name
-----------------------------------------	------

Element Definition Text

A character string that modifies a class word. It is
normally associated with quantities.

Domain Definition Text

A generic domain comprised of the following ASCII
characters: A-Z; Hyphen (-); and Underscore (_).

Record Number: 6

```
-----  
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth  
-----  
          6      1      Elm-Dtype-Cat      QL          16
```

Data Element Name

Information Element Data Type Category

Element Definition Text

The editing type of the data values associated with the
element.

Domain Definition Text

A generic domain comprised of the following qualitative
data values: Bit-String; Integer; Character-String;
Fixed-Point; Floating-Point.

Record Number: 7

```
-----  
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth  
-----  
          7      1      Elm-Def-Txt      QL          468
```

Data Element Name

Information Element Definition Text

Element Definition Text

Narrative describing the organizational context or the meaning of an information element.

Domain Definition Text

A generic domain comprised of the characters in the ASCII graphic character set.

Record Number: 8

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Lngh
8	1	Elm-Dom-Def-Txt	QL	468

Data Element Name

Information Element Domain Definition Text

Element Definition Text

Narrative describing the acceptable set of data values for a specific information element.

Domain Definition Text

A generic domain comprised of the characters in the ASCII
graphic character set.

Record Number: 9

De_Count Ver Nbr Mnemonic ID Data Val ID Lngth

9 1 Elm-Cmt-Txt QL 468

Data Element Name

Information Element Comment Text

Element Definition Text

An administrative narrative regarding a reference
element, data element or data element alias.

Domain Definition Text

A generic domain comprised of the characters in the ASCII
graphics character set.

Record Number: 10

```
-----
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth
-----
      10      1      Elm-Max-Dv-Lgthchr  QN          4
```

Data Element Name

Information Element	Maximum Data Value	Length
---------------------	--------------------	--------

Characters

Element Definition Text

The maximum number of characters an information element data value may contain.

Domain Definition Text

A specific domain of quantitative data values ranging from 0001 to 9999.

Record Number: 11

```
-----
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth
-----
      11      1      Elm-Just-Cat      QN          7
```

Data Element Name

Information Element Justification

Category

Element Definition Text

The positional justification of data values within a storage field.

Domain Definition Text

A specific domain comprised of the following qualitative data values: Left; Right.

Record Number: 12

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Lngh
12	1	Elm-Std-Auth-ID	QL	4

Data Element Name

Information Element Standardization Authority

Identifier

Element Definition Text

The branch of service, Government, or international organization that approved the element.

Domain Definition Text

A specific domain comprised of the following qualitative data values: ANSI, NIST, ISO, DA, OJCS, NATO, DOD.

Record Number: 13

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Length
13	1	Elm-Adoc-Nme	QL	30

Data Element Name

Information Element Authorization Document

Name

Element Definition Text

A character-string given to the document (regulation, publication, document, or other) that authorizes a reference or data element.

Domain Definition Text

A generic domain comprised of the following ASCII characters: A-Z; Hyphen (-); Underscore (_); Point (.); and 0-9.

Record Number: 14

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Lngh
14	1	Elm-Dv-List-Txt	QL	312

Data Element Name

Information Data Value Source List Text

Element Definition Text

The source in which lengthy code lists are enumerated for the user. This source can either be a manual or automated medium.

Domain Definition Text

A generic domain comprised of the characters in the ASCII graphic character set.

Record Number: 15

```
-----
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth
-----
      15      1      Elm-Dv-Nme          QL          35
```

Data Element Name

```
-----
      Information Element Data Value      Name
```

Element Definition Text

```
-----
An occurrence of a character string given to an
acceptable data value.
```

Domain Definition Text

```
-----
A generic domain comprised of the following ASCII
characters: A-Z, -, ., 0-9, /, _, &.
```

Record Number: 16

```
-----
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth
-----
      16      1      Elm-Dv-Def-Txt      QL          200
```

Data Element Name

Information Element Data Value Definition Text

Element Definition Text

Narrative describing the meaning of a specific data value.

Domain Definition Text

A generic domain of the characters in the ASCII graphic character set.

Record Number: 17

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Lngh
----------	---------	-------------	-------------	------

-----	-----	-----	-----	-----
-------	-------	-------	-------	-------

17	1	Qn-Hrng-Nbr	QL	15
----	---	-------------	----	----

Data Element Name

Information Quantitative Data High-Range

Number

Element Definition Text

A character string indicating the largest value for quantitative data, when a domain set is expressed as a range of acceptable values.

Domain Definition Text

A generic domain comprised of the ASCII characters: 0-9, point (.), and minus (-).

Record Number: 18

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Length
18	1	Qn-Lrng- Nbr	QL	15

Data Element Name

Information Quantitative Data Low-Range

Number

Element Definition Text

A character string indicating the smallest value for quantitative data when, when a domain set is expressed as a range of acceptable values.

Domain Definition Text

A generic domain comprised of the ASCII characters 0-9,
point (.), and minus (-).

Record Number: 19

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Length
19	1	Qn-Scale-Nbr	QL	2

Data Element Name

Information Quantitative Data Scale Number

Element Definition Text

A character string indicating the integer that determines
the decimal point placement in an element for a fixed
point data type.

Domain Definition Text

A generic domain comprised of the ASCII characters 0-9
and decimal point (.).

Record Number: 20

```
-----
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth
-----
      20      1      Qn-Dv-Nbr          QL          35
```

Data Element Name

Information Quantitative Data Value Number

Element Definition Text

A character string which represents the numerical values used when mathematical operations must be performed on qualitative data.

Domain Definition Text

A generic domain comprised of the ASCII characters 0-9 and decimal point (.).

Record Number: 21

```
-----
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth
-----
      21      1      De-Nme          QL          250
```

Data Element Name

Information Data Element Name

Element Definition Text

A character string given to a data element based on the prime term and a reference element name.

Domain Definition Text

A generic domain comprised of the ASCII characters A-Z, Hyphen (-), and Underscore (_).

Record Number: 22

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Lngh
22	1	Pwor lme	QL	27

Data Element Name

Information Prime Word Name

Element Definition Text

A character string in a data element name that represents the data grouping to which the data element belongs.

Domain Definition Text

A specific domain comprised of the qualitative data values listed in Appendix J (Figure J-2) of AR 25-9.

Record Number: 23

De_Count Ver Nbr Mnemonic ID Data Val ID Lngth

23 1 De-Mnmnic-Abb QL 18

Data Element Name

Information Data Element Mnemonic Identifier
Element Definition Text

A short form of data element character string.

Domain Definition Text

A generic domain comprised of the ASCII characters A-Z, Hyphen (-), Underscore (_), slash (/), left Paren "(", and right Paren ")".

Record Number: 24

```
-----  
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth  
-----  
          24      1      De-resp-Ofc-Nme      QL          250
```

Data Element Name

Information Data Element Responsible Office

Name

Element Definition Text

A character string given to the office and/or person designated by the information class proponent as the functional expert for defining, reviewing, and updating a data element and its attributes.

Domain Definition Text

A generic domain comprised of the ASCII characters A-Z, Hyphen (-), Underscore (_), point (.), and 0-9.

Record Number: 25

```
-----
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth
-----
      25      1      De-Secu-Cat      QL           25
```

Data Element Name

```
-----
Information Data Element Security      Category
Element Definition Text
```

```
-----
The level of security required by the realization of a
data element's structure and data values in some physical
media or representation.
```

Domain Definition Text

```
-----
A specific domain comprised of the qualitative data
values listed under the definition of Qualitative data
values for De-Secu-Cat.
```

Record Number: 26

```
-----
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth
-----
      26      1      Ic-Id           QL           3
```

Data Element Name

Information Class	Identifier
-------------------	------------

Element Definition Text

An indication of the information class within the current HQDA Information Model.

Domain Definition Text

A specific domain comprised of qualitative data values ranging from 001-066. See Appendices C and J for a list of information classes which corresponds to these identifiers.

Record Number: 27

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Lngh
----------	---------	-------------	-------------	------

-----	-----	-----	-----	-----
-------	-------	-------	-------	-------

27	1	Ic-Nme	QL	40
----	---	--------	----	----

Data Element Name

Information Class	Name
-------------------	------

Element Definition Text

A character string given to the class of information to which a data element is assigned in accordance with the current HQDA Information Model.

Domain Definition Text

A specific domain comprised of the qualitative data values listed in Appendix C and J of AR 25-9.

Record Number: 28

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Lngh
28	1	Daarch-Sa-Nme	QL	30

Data Element Name

Information Army Data Architecture Subject-Area
Name

Element Definition Text

A character string given to an Army data architecture subject-area.

Domain Definition Text

A specific domain comprised of the qualitative data values listed in Appendix D of AR 25-9.

Record Number: 29

De_Count Ver Nbr Mnemonic ID Data Val ID Lngth

29 1 Ic-Propnt-Nme QL 10

Data Element Name

Information Class Proponent Name

Element Definition Text

A character string which designates an organization that has been assigned responsibility for an information class within the current HQDA Information Model.

Domain Definition Text

A specific domain comprised of the qualitative data values listed in Appendix C of AR 25-9.

Record Number: 30

```
-----  
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth  
-----  
          30      1      In-Proc-Nme        QL          50
```

Data Element Name

```
-----  
Information Process      Name
```

Element Definition Text

```
-----  
A designation for an object expressed in a word or words.
```

Domain Definition Text

```
-----  
A specific domain comprised of the qualitative data  
values listed in Appendix C of AR 25-9.
```

Record Number: 31

```
-----  
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth  
-----  
          31      1      De-Timness-ID      QL           3
```

Data Element Name

Information Data Element Timeliness

Identifier

Element Definition Text

An indication of how often data values must be updated.

Domain Definition Text

A specific domain comprised of qualitative values found
in the "Qualitative Data Values name and definition
"section - next page.

Record Number: 32

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Lngh
----------	---------	-------------	-------------	------

-----	-----	-----	-----	-----
-------	-------	-------	-------	-------

32	1	Ql-Dv-Acry-Nbrpct	QL	6
----	---	-------------------	----	---

Data Element Name

Information Qualitative Data Value Accuracy

Number Percent

Element Definition Text

- character string indicating how accurate a qualitative data value must be.

Domain Definition Text

A specific domain comprised of qualitative data values 0-9 and a point (.) expressing a percent range ranging from 0.01 to 100.00.

Record Number: 33

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Lngh
33	i	Qn-Acry-ID	QL	2

Data Element Name

Information Quantitative Data Accuracy

Identifier

Element Definition Text

A character string indicating how accurate a quantitative data value must be.

Domain Definition Text

A specific domain comprised of the qualitative data values found in the "Qualitative Data Value names and definitions" section - on the next page.

Record Number: 34

De_Count Ver Nbr Mnemonic ID Data Val ID Lngth

34 1 De-Calc-Fmla-Txt QL 250

Data Element Name

Information Data Element Calculation Formula
Text

Element Definition Text

Narrative expressing the algorithmic formula for a data element that is derived.

Domain Definition Text

A generic domain comprised of the characters in the ASCII graphic character set.

Record Number: 35

```
-----
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth
-----
          35      1      Dea-Nme           QL           250
```

Data Element Name

Information Data Element Alias	Name
--------------------------------	------

Element Definition Text

A character string given to a nonstandard data element.

Domain Definition Text

A generic domain comprised of the ASCII characters A-Z;
Hyphen(-); Underscore (_); Point (.), and 0-9.

Record Number: 36

```
-----
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth
-----
          36      1      Dea-Host-App-Nme  QL           100
```

Data Element Name

Information Data Element Alias Host Application

Name

Element Definition Text

A character string given to an application/program that contains a data element alias.

Domain Definition Text

A generic domain comprised of the ASCII characters A-Z; Hyphen (-); Underscore (_); Point (.); and 0-9.

Record Number: 37

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Lngh
37	1	Dea-Host-Sys-Nme	QL	100

Data Element Name

Information Data Element Alias Host System Name

Element Definition Text

A character string given to an information system on which the application/program that contains a data element alias runs.

Domain Definition Text

A generic domain comprised of the ASCII characters A-Z; Hyphen (-); Underscore (_); Point (.); and 0-9.

Record Number: 38

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Length
33	1	Dea-Resp-Ofc-Nme	QL	250

Data Element Name

Information Data Element Alias Responsible

Office Name

Element Definition Text

A character string given to the office and/or person designated by the information class proponent as the functional expert for defining, reviewing, and updating a

data element alias and its attributes.

Domain Definition Text

A generic domain comprised of the ASCII characters A-Z;
Hyphen (-), Underscore (_); Point (.); and 0-9.

Record Number: 39

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Lngh
39	1	Dea-Int-Fmt-Cat	QL	12

Data Element Name

Information Data element Alias Internal Format

Category

Element Definition Text

The internal storage format of a data element alias on
the parent host/application.

Domain Definition Text

A specific domain comprised of following qualitative data
values: ASCII, EBCDIC, BINARY, DECIMAL, PACKED,
COMPRESSED, DOUBLE.

Record Number: 40

```
-----  
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth  
-----  
          40      1      Elm-Ver-Nbr      QL          5
```

Data Element Name

Information Element Version Number

Element Definition Text

An increasing ordinal representation of changes to a
standard element.

Domain Definition Text

A generic domain comprised of the ASCII characters 0-9
and point (.) in the format VV.MM where VV can be a
positive integer value and MM can be a positive integer
value. VV registers the current version number and MM

Record Number: 41

```
-----  
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth  
-----  
          41      1      Elm-Status-ID      QL          2
```

Data Element Name

Information Element Status

Identifier

Element Definition Text

An indication of the current status of a reference or data element in relation to the standardization process.

Domain Definition Text

A specific domain comprised of the following qualitative data values: PR=PROPOSED; CA=CANDIDATE; AF=APPROVED FUNCTIONALLY; AP=APPROVED; IN=INSTALLED; AR=ARCHIVED.

Record Number: 42

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Length
----------	---------	-------------	-------------	--------

-----	-----	-----	-----	-----
-------	-------	-------	-------	-------

42	1	Elm-Cntr-ID	QL	9
----	---	-------------	----	---

Data Element Name

Information Element Counter

Identifier

Element Definition Text

A unique number assigned to each reference element and data element.

Domain Definition Text

A specific domain comprised of a set of unique numbers assigned to a reference or data element.

Record Number: 43

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Length
43	1	Tech-Rev-Status-ID	QL	2

Data Element Name

Information Technical Review Status

Identifier

Element Definition Text

The status of an element after the technical review has been performed.

Domain Definition Text

A specific domain comprised of the following qualitative data values: DT=DISAPPROVED TECHNICALLY; AT=APPROVED TECHNICALLY.

Record Number: 44

De_Count Ver Nbr Mnemonic ID Data Val ID Lngth

44 1 Elm-Appvl-Date QN 8

Data Element Name

Information Element Approval Date

Element Definition Text

The date a standard element is approved as an Army standard.

Domain Definition Text

A specific domain of quantitative data values ranging from 19890627 to 20201231.

Record Number: 45

```
-----
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth
-----
      45      1      Elm-Mod-Date      QN           8
```

Data Element Name

```
-----
      Information Element Modification      Date
```

Element Definition Text

```
-----
The date a change to a standard element is approved.
```

Domain Definition Text

```
-----
A specific domain of quantitative data values ranging
from 19890627 to 20201231.
```

Record Number: 46

```
-----
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth
-----
      46      1      Elm-Instln-Date      QN           8
```

Data Element Name

```
-----
      Information Element Installation      Date
```

Element Definition Text

The date designated by a responsible Army authority,
after which use of the standard information element is a
mandatory requirement in support of all Army information
exchange requirements within the scope of AR 25-9.

Domain Definition Text

A specific domain of quantitative data values ranging
from 19890627 to 20201231.

Record Number: 47

De_Count Ver Nbr Mnemonic ID Data Val ID Lngth

47 1 Elm-Archvl-Date QN 8

Data Element Name

Information Element Archival Date

Element Definition Text

The date designated by a responsible Army Authority,
after which use of the standard information element is no
longer required in support of all Army information

exchange requirements within the scope of AR 25-9.

Domain Definition Text

A specific domain of quantitative data values ranging
from 19890627 to 20201231.

Record Number: 48

De_Count	Ver Nbr	Mnemonic ID	Data Val ID	Lngh
48	1	Elm-ID	QL	1

Data Element Name

Information Element Type	Identifier
--------------------------	------------

Element Definition Text

An indication of the class of element.

Domain Definition Text

A specific domain comprised of the following qualitative
data values: R=REFERENCE ELEMENT; D=DATA ELEMENT.

Record Number: 49

```
-----  
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth  
-----  
49        1      Elm-Rev-Act-ID      QL           1
```

Data Element Name

```
-----  
Information Element Review Action  Identifier  
Element Definition Text
```

```
-----  
The documented decision of a responsible Army authority  
to approve or disapprove a reference or data element.
```

Domain Definition Text

```
-----  
A specific domain comprised of the following qualitative  
data values:  A=APPROVED; D=DISAPPROVED.
```

Record Number: 50

```
-----  
De_Count  Ver Nbr  Mnemonic ID      Data Val ID  Lngth  
-----  
50        1      Elm-Revr-Cmt-Txt    QL           9999
```

Data Element Name

Information Element Reviewer Comment Text

Element Definition Text

A narrative which provides remarks pertinent to the
evaluation of a candidate element.

Domain Definition Text

A generic domain comprised of the characters in the
ASCII graphic character set.

APPENDIX F

1. APWTRELM: Reference Element

The key field is `Relm_counter` and the foreign property that constitutes a 1:N link from `APWTCWOR` to `APWTRELM` is `Elm_clwd_nme`. `Relm_counter` participates as a foreign key and 1:N relationships with `APWTREFD`, `APWTREQD`, and `APWTDE`.

2. APWTREQD: Reference Element Data Value Number

The primary key field is `Elm_dv_nme` and the foreign key is `Relm_counter`. `APWTREQD` is the "N" in a 1:N relationship with `APWTRELM`.

3. APWTREFD: Reference Element Item

The primary key field is `Qn_dv_nbr` and the foreign key is `Relm_counter`. `APWTREQD` is the "N" in a 1:N relationship with `APWTRELM`.

4. APWTCWOR: Class Word

The key field is `Elm_clwd_nme` and is the "1" in a 1:N relationship with `APWTRELM`.

5. APWTDE: Data Element

The key field is `De_counter`. Foreign properties that provide a link are `Relm_counter`, `Prwd_nme`, and `Ic_id`.

APWTDE acts as the "N" with APWTPWOR and APWTIC and acts as the "1" with APWTDEQD and APWTDEDI.

6. APWTDEQD: Data Element Data Value Number

The primary key field is Qn_dv_nbr and the foreign key is De_counter. APWTDEQD is the "N" in a 1:N relationship with APWTDE.

7. APWTDEDI: Data Element Data Item

The primary key field is Elm_dv_nme and the foreign key is De_counter. APWTDEDI is the "N" in a 1:N relationship with APWTDE.

8. APWTAL: Alias Element

The key field is Al_counter. De_counter is a foreign property constitutes a 1:N link with APWTDE. Al_counter is the foreign key for and acts as the "1" in a 1:N relationship with APWTALDI, APWTALQD, and APWTSS.

9. APWTALQD: Alias Data Value Number

The primary key is Qn_dv_nbr and the foreign key is Al_counter. APWTALQD acts as the "N" in a 1:N relationship with APWTAL.

10. APWTALDI: Alias Data Value Name

The primary key is Elm_dv_nme and the foreign key is Al_counter. APWTALDI acts as the "N" in a 1:N relationship with APWTAL.

11. APWTSS: Alias Host System

The primary key is Dea_host_app_nme and the foreign key is Al_counter. APWTSS acts as the "N" in a 1:N relationship with APWTAL.

12. APWTPWOR: Prime Word

The key field is Prwd_nme and is the "1" in a 1:N relationship with APWTDE. The Daarch_sa_nme attribute acts as a foreign property link and 1:N relationship with APWTIC.

13. APWTIC: Information Class

The key field is Ic_id. Daarch_sa_nme is a foreign property and link in a 1:N relationship with APWTPWOR. Ic_id also is a foreign property within APWTDE and acts as the "1". APWTIC acts as the "1" in a 1:N relationship with APWTDE.

APPENDIX G

This appendix shows form and report materializations for the dictionary prototype. Starting on the following page, these materializations start with the Introduction/Password screen and follow the following order: Main Menu Help screen, Reference Element options, Data Element options, Alias Element options, Class Word options, Prime Word options.

A. Introduction/Password and Main Menu Help Screens

Password:

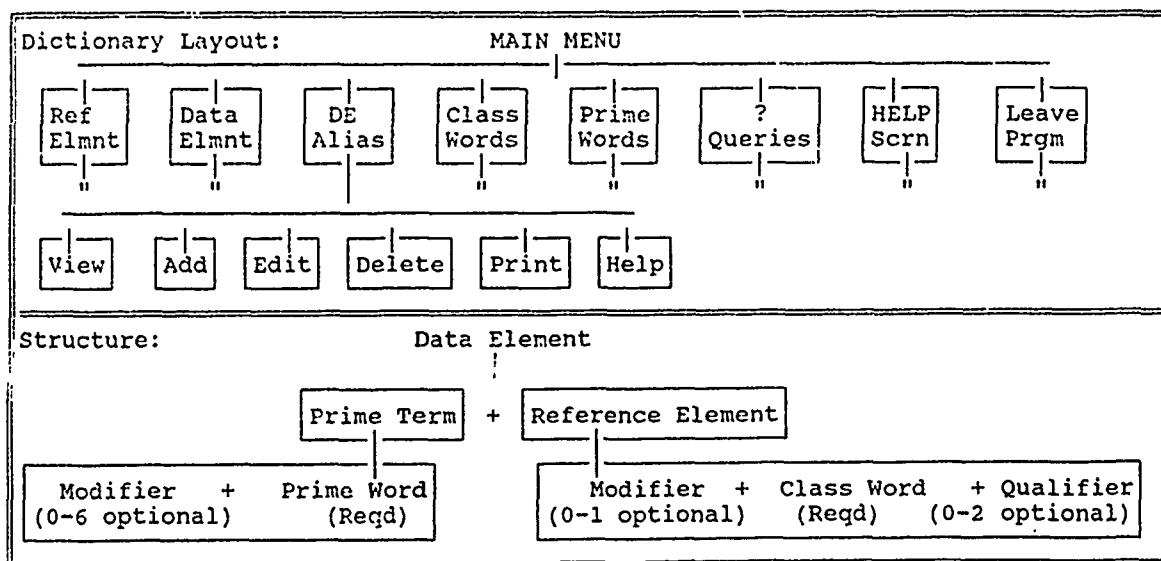
Enter password for the application; [Esc] to cancel; [Enter] for no password.

Department of Defense
(DOD)

Standardized Data Element Dictionary
and
Desktop Glossary

By
Dr. Dan Dolk & Capt J. S. Bacheller
Naval Postgraduate School
Monterey, CA
19900612

Press any key to end the help



B. Reference Element Options

Press any key to end the help

REFERENCE ELEMENT MENU HELP SCREEN

List Reference Elmts: View a list of all Ref Elements (displays: Record number, Ref Elmnt Number, Type(QL/QN, Status, Name).

Add Reference Elmts : Sends you to a lower menu where you have the option to add either Qualitative or Quantitative Ref Elmts.

Edit Reference Elmts: Sends you to a lower menu like the "Add" choice does. You may view or edit selected (by Number) Ref Elmts. The number is "display only" to maintain the data chain on that particular number.

Delete Reference Ele: You may delete a Reference Element, but first must delete the one-to-many attributes related to it first.

Print Reports : Sends you to a lower menu where you may select from numerous reports to print.

Help : Gives you this screen.

Press [F2] when finished viewing the table
Total records: 70

GENERAL REFERENCE ELEMENT LIST FORM

Rec#	Ref#	Data-Type	Status	Reference Element Name
1	1	QL	PR	Category
2	2	QN	PR	Acceleration
3	3	QL	PR	Code
4	4	QN	PR	Amount
5	5	QL	PR	Name

Press the ARROW keys to scroll through the values.

[F2] - Data entry completed, Esc - Cancel data entry, Ctrl-U - Undo last change

REFERENCE ELEMENT CREATE FORM			Rec Num: 1
Where "[F1]" is indicated, press the key [F1] for a list of choices.			
Reference Element Nbr:	Version Nbr:	Element Status ID:	[F1]
Modifier Name:	Class Word Name:	Qualifiers 1&2	[F1]
Resulting Reference Element Name			
Data Value Type:		Max Length Characters:	
Data Type Category:	[F1]	Justification Category:	
Std Authority ID:	[F1]	Auth Doc Name:	
Definition Text:			
Press PgDn			

[F2] - Data entry completed, Esc - Cancel data entry, Ctrl-U - Undo last change

Comment Text:		
Domain Definition Text:		
Data Value Source List Text:		
Creator ID:	Element Appvl Date:	(YYYYMMDD)
Review Status:	Element Mod Date:	(YYYYMMDD)
Press PgUp or PgDn		

[F2] - Data entry completed, Esc - Cancel data entry, Ctrl-U - Undo last change

◀ Complete for Qualitative Reference Elements only. Otherwise, press PgDn to create Quantitative Reference Elements.

Press [F3] in order to move the cursor in/out of the fields below.
Use the ARROW keys to scroll through the values.

Rec # Data Value: Data Value Definition Text:

Press PgUp or PgDn

[F2] - Data entry completed, Esc - Cancel data entry, Ctrl-U - Undo last change

Complete for Quantitative Reference Elements only. Otherwise, press PgUp to create Qualitative Reference Elements.

Low Range Number:
High Range Number:

◀ Quan Data Scale Number:

Press [F3] in order to move the cursor in/out of the fields below.
Use the ARROW keys to scroll through the values.

Rec # Data Value Nbr: Data Value Definition Text:

Press PgUp

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change

REFERENCE ELEMENT EDIT FORM		Rec Num: 1
Where "[F1]" is indicated, press the key [F1] for a list of choices.		
Reference Element Nbr:	1	Version Nbr: 1 ◀ Element Status ID: PR [F1]
Modifier Name:	Class Word Name:	Qualifiers 1&2
	Category	[F1]
Resulting Reference Element Name Category		
Data Value Type:	QL	Max Length Characters: 250
Data Type Category:	Character-string [F1]	Justification Category:
Std Authority ID:	DA [F1]	Auth Doc Name:
Definition Text:		
A division or subset in a system of classification in which all items share the same concept.		
Press PgDn		

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change

Comment Text:		
The ASCII graphic character set is in DA Pam 25-DM.		
Domain Definition Text:		
A generic domain comprised of the characters in the ASCII graphic character set.		
Data Value Source List Text:		
Creator ID:	ODISC4	Element Appvl Date: (YYYYMMDD)
Review Status:		Element Mod Date: (YYYYMMDD)
Press PgUp or PgDn		

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change

◀ Complete for Qualitative Reference Elements only. Otherwise, press PgDn to create Quantitative Reference Elements.

Press [F3] in order to move the cursor in/out of the fields below.
Use the ARROW keys to scroll through the values.

Rec # Data Value: Data Value Definition Text:

Press PgUp or PgDn

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change

Complete for Quantitative Reference Elements only. Otherwise, press PgUp to create Qualitative Reference Elements.

Low Range Number:
High Range Number:

◀ Quan Data Scale Number:

Press [F3] in order to move the cursor in/out of the fields below.
Use the ARROW keys to scroll through the values.

Rec # Data Value Nbr: Data Value Definition Text:

Press PgUp

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change
[Del] - Delete a record

QUALITATIVE/QUANTITATIVE REFERENCE ELEMENT DELETE FORM

Rec Num: 1

Reference Element Number: 1 Version Nbr: 1 Element Status ID: PR

Modifier Name:

Class Word Name: Category

Qualifier Name :

Qualifier Name :

Reference Element Name _____

Category _____

Use the [PgDn] key to locate the Reference Element desired.

To delete either a Qualitative or Quantitative Reference Element, you must go to pages 2 and 3 in order to first delete the one-to-many attributes (using the [Del] key on each record). These must all be purged prior to deleting the actual Reference Element (Parent Record).

Next, return to page 1 of the Reference Element and press [Del].

Press PgDn

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change
[Del] - Delete a record

[illegible]

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change
[Del] - Delete a record

◀ Press [F3] to move the cursor in/out of the fields below. Use the
ARROW keys to scroll through the values.

Rec #	Data Value:	Data Value Definition Text:
-------	-------------	-----------------------------

Press PgUp

List of Reference Elements

7/23/90
Page 1

Ref Elmt #	Data Val Type ID	Status ID	Reference Element Name
1	QL	PR	Category
2	QN	PR	Acceleration
3	QL	PR	Code
4	QN	PR	Amount
5	QL	PR	Name
6	QL	PR	Number
7	QL	PR	Number Percent
8	QL	PR	Text
9	QL	PR	Identifier
10	QL	PR	Date-Time-Group
11	QN	PR	Time
12	QN	PR	Year
13	QN	PR	Date
14	QN	PR	Year-Month Date
15	QN	PR	Ordinal Date
16	QN	PR	Latitude
17	QN	PR	Latitude Seconds

Reference Element Report

7/23/90
Page 1

Record Number: 1

Reference Element Number: 1 Version Nbr: 1 Status ID: PR

Reference Element Name:
Category

Data Value Type ID: QL Max Length Characters: 250
Data Type Category: Character-string Justification Category:

Standardization Authority ID: DA
Authorization Document Name:

Creator ID: ODISC4 Approval Date: (YYYYMMDD)
Review Status: Mod Date: (YYYYMMDD)

If Quantitative: Low Range Nbr:
High Range Nbr:
Scale Nbr:

Definition Text:
A division or subset in a system of classification in which all items
share the same concept.

Comment Text:
The ASCII graphic character set is in DA Pam 25-DM.

Domain Definition Text:
A generic domain comprised of the characters in the ASCII graphic
character set.

Source List text:

Qualitative Reference Element
Data Values

7/23/90
Page 1

Record Number: 1

Reference Element Number: 56 Version Nbr: 1 Status ID: PR

Reference Element Name:
Angle

Data Value Type: QN Max Length Characters: 15

Rec	Data Value	Data Value
Nbr	Name	Definition Text

1	Acute	Acute angle
---	-------	-------------

2	Obtuse	Obtuse angle
---	--------	--------------

Quantitative Reference Element
Data Values

7/23/90
Page 1

Record Number: 1

Reference Element Number: 3 Version Nbr: 1 Status ID: PR

Reference Element Name:
Code

Data Value Type: QL Max Length Character: 35

Rec	Data Value	Data Value
Nbr	Number	Definition Text

1	Letters	Letters for code.
---	---------	-------------------

2	Numbers	Numbers used for code.
---	---------	------------------------

C. Data Element Options

Press any key to end the help

HELP SCREEN	
LIST Data Elements	- Provides a view of a list of Data Elements (Rec Nbr, Data Element Nbr, Data Value Type, and Name).
ADD Data Elements	- Add Qualitative/Quantitative Elements. Creation comes from adding a Reference Element to a Prime Term. The Data Element "takes-on" the Data Value of the Reference Element (Qualitative(QL)/Quantitative(QN)).
EDIT Data Elements	- Edit a Data Element that you select by entering the Data Element Number when prompted.
Del Data Elements	- After selecting this choice, page-through the Data Elements to locate a record. First delete all full fields under the "[F3]" prompts (they come from different tables). Then simply press "[Del]" when at the beginning of the desired record.
Print Data Elements	- Moves to a lower menu for print options.
HELP	- This screen.

Press [F2] when finished viewing the table
Total records: 50

GENERAL DATA ELEMENT LIST FORM				
Rec#	DE#	Data Type ID	Data Element Name	Name
1	1	QL	Information Reference Element	
2	2	QL	Information Data Value Type Identifier	
3	3	QL	Information Element Class Word	Name

Press the ARROW keys to scroll through the values.

[F2] - Data entry completed, Esc - Cancel data entry, Ctrl-U - Undo last change

QUANTITATIVE/QUALITATIVE DATA ELEMENT CREATE FORM			Rec #: 1
Where [F1] is shown, press the [F1] key to select from a list of choices.			
Data Element Number:	Version Number:	Status ID: [F1]	
"Before" Modifiers	Prime Word	"After" Modifiers	
1	[F1]	7	
2		8	
3		9	
4		10	
5		11	
6		12	
<div style="border: 1px solid black; display: inline-block; padding: 5px;"> NOTE: Use an option- 6 Modifiers max (to a Data Element). </div>			
Ref Elmnt Nbr:	[F1]	Name:	
Data Value ID:			
Resulting Data Element Name			
Press PgDn			

[F2] - Data entry completed, Esc - Cancel data entry, Ctrl-U - Undo last change

Mnemonic Abbr:	Information Class ID:	[F1]
Security Cat:	[F1]	Max Length Character:
Standardization Authority ID:	[F1]	Timeliness ID:
Authorization Document Name:		[F1]
Creator ID:	Approval Date:	(YYYYMMDD)
Admin Rvw Status:	Modification Date:	(YYYYMMDD)
Press [F3] to move in/out of the fields below.		
Calculation Fmla Text:		
Source List Text:		
Comment Text:		
Domain Definition Text:		
Press PgUp or PgDn		

Responsible
Office Name:

Definition
Text:

For Qualitative Data Elements Only (QL)

Qualitative Data Value Accuracy Number Percent:

—Press [F3] to move in/out of the fields below. Use ARROW keys to scroll.—

Rec#	Data Value Name	Data Value Definition Text
------	-----------------	----------------------------

Press PgUp or PgDn

```

      For Quantitative Data Elements Only (QN)

Quantitative Data Accuracy Identifier:      <[F1]
Quantitative Scale Number:
Quantitative Low Range Number:
Quantitative High Range Number:

—Press [F3] to move in/out of the fields below. Use ARROW keys to scroll.—
Rec#      Data Value Number              Data Value Definition Text

```

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change

QUANTITATIVE/QUALITATIVE DATA ELEMENT EDIT FORM				Rec #: 1
Where [F1] is shown, press the [F1] key to select from a list of choices.				
Data Element Number: 1		Version Number: 1 ◀	Status ID: PR [F1]	
Modifiers	Prime Word		Modifiers	
1	Information [F1]		7Reference	
2			8Element	
3			9	
4			10	
5			11	
6			12	
<div style="border: 1px solid black; padding: 5px; display: inline-block;"> NOTE: Use an option- 6 Modifiers max (to a Data Element). </div>				
Ref Elmnt Nbr: 5 [F1]		Name: Name		
Data Value ID: QL*				
Resulting Data Element Name				
Information		Reference	Element	Name
Press PgDn				

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change

Mnemonic Abbr: Relm-Nme ◀		Information Class ID: 1 [F1]
Security Cat: Unclassified [F1]		Max Length Character: 80
Standardization Authority ID: DA [F1]		Timeliness ID: Z [F1]
Authorization Document Name:		
Creator ID: ODISC4	Approval Date: (YYYYMMDD)	
Admin Rvw Status:	Modification Date: (YYYYMMDD)	
Press [F3] to move in/out of the fields below.		
Calculation		
Fmla Text:		
Source List		
Text:		
Comment A reference element name consists of an optional modifier (M), a		
Text: class word (cw), and one or two optional qualifiers (Q). For		
example, horizontal velocity miles-per-hour; static pressure		
Domain A generic domain comprised of the following ASCII characters:		
Definition A - Z; Hyphen (-); and Underscore (_).		
Text:		
Press PgUp or PgDn		

Responsible
Office Name

Definition A character string given to a reference element based on a
Text: class word that identifies a domain. (See comment text).

For Qualitative Data Elements Only (QL)

Qualitative Data Value Accuracy Number Percent:

—Press [F3] to move in/out of the fields below. Use ARROW keys to scroll.—

Rec#	Data Value Name	Data Value Definition Text
------	-----------------	----------------------------

Press PgUp or PgDn

```

      For Quantitative Data Elements Only (QN)
Quantitative Data Accuracy Identifier:      <[F1]
Quantitative Scale Number:
Quantitative Low Range Number:
Quantitative High Range Number:

—Press [F3] to move in/out of the fields below. Use ARROW keys to scroll.—
Rec#      Data Value Number              Data Value Definition Text

```

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change
 [Del] - Delete a record

QUANTITATIVE/QUALITATIVE DATA ELEMENT DELETE FORM				Rec #: 1
To delete an entire Data Element, you must first delete all filled fields below the [F3] prompts on pages 2-4 (using the [Del] key). When the fields under [F3] are empty, press the [Del] key.				
Data Element Number:	1	Version Number:	1	Status ID: PR [F1]
Modifiers		Prime Word		Modifiers
1		Information [F1]	7	Reference
2			8	Element
3			9	
4			10	
5			11	
6			12	
Ref Elmnt Nbr:	5 [F1]	Name:	Name	
Data Value ID:	QL*			
Resulting Data Element Name				
		Information Reference Element	Name	
Press PgDn				

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change
 [Del] - Delete a record

Mnemonic Abbr: Relm-Nme	1	Information Class ID:	1	[F1]
Security Cat: Unclassified		[F1] Max Length Character:	80	
Standardization Authority ID:	DA [F1]	Timeliness ID:	2	[F1]
Authorization Document Name:				
Creator ID:	ODISC4	Approval Date:	(YYYYMMDD)	
Admin Rvw Status:		Modification Date:	(YYYYMMDD)	
Press [F3] to move in/out of the fields below.				
Calculation				
Fmla Text:				
Source List				
Text:				
Comment	A reference element name consists of an optional modifier (M), a			
Text:	class word (cw), and one or two optional qualifiers (Q). For			
	example, horizontal velocity miles-per-hour; static pressure			
Domain	A generic domain comprised of the following ASCII characters:			
Definition	A - Z; Hyphen (-); and Underscore (_).			
Text:				
Press PgUp or PgDn				

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change
[Del] - Delete a record

Responsible
Office Name:

Definition A character string given to a reference element based on a
Text: class word that identifies a domain. (See comment text).

For Qualitative Data Elements Only (QL)

Qualitative Data Value Accuracy Number Percent:

—Press [F3] to move in/out of the fields below. Use ARROW keys to scroll.—

Rec#	Data Value Name	Data Value Definition Text
------	-----------------	----------------------------

Press PgUp or PgDn

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change
[Del] - Delete a record

For Quantitative Data Elements Only (QN)

Quantitative Data Accuracy Identifier: ◀[F1]

Quantitative Scale Number:

Quantitative Low Range Number:

Quantitative High Range Number:

—Press [F3] to move in/out of the fields below. Use ARROW keys to scroll.—

Rec#	Data Value Number	Data Value Definition Text
------	-------------------	----------------------------

Press PgUp

Data Element List

7/23/90
Page 1

Rec Nbr	Data Elmnt #	Data Val Type	Data Element Name	
1	1	QL	Information Reference Element	Name
2	2	QL	Information Data Value Type	Identifier
3	3	QL	Information Element Class Word	Name
4	4	QL	Information Element Modifier	Name
5	5	QL	Information Reference Element Qualifier	Name
6	6	QL	Information Element Data Type	Category
7	7	QL	Information Element Definition	Text
8	8	QL	Information Element Domain Definition	Text
9	9	QL	Information Element Comment	Text
10	10	QN	Information Element Maximum Data Value	Length Characters
11	11	QL	Information Element Justification	Category

Data Element Report

7/23/90
Page 1

Record Number: 1

Data Element Number: 1 Version Nbr: 1 Status ID: PR

Data Element Name:
Information Reference Element Name

Data Value Type ID: QL Max Length Characters: 80
Mnemonic Abbreviation: Relm-Nme Information Class ID: 1
Security Category: Unclassified
Timeliness ID: Z

Standardization Authority ID: DA
Authorization Document Name:

Creator ID: ODISC4 Element Approval Date: (YYYYMMDD)
Review Status: Element Mod Date: (YYYYMMDD)

Responsible Office Name:

Definition Text:
A character string given to a reference element based on a class word that identifies a domain. (See comment text).

Domain Definition Text:
A generic domain comprised of the following ASCII characters: A - Z; Hyphen (-); and Underscore (_).

Comment Text:
A reference element name consists of an optional modifier (M), a class word (cw), and one or two optional qualifiers (Q). For example, horizontal velocity miles-per-hour; static pressure millibars.

Calculation Formula Text:

Source List Text:

If a Qualitative Data Value: Qual Data Value Accrcy %:

If a Quantitative Data Value:
Quan Low Range Number: Quan Data Accrcy ID:
Quan High Range Number: Quan Scale Number:

Qualitative Data Element
Data Values

7/23/90
Page 1

Record Number: 1

Data Element Number: 2 Version Nbr: 1 Status ID: PR

Data Element Name:
Information Data Value

Type Identifier

Data Value type: QL

Rec Data Value
Nbr: Name:

Data Value
Definition Text:

1 QL

Qualitative Data Value

2 QN

Quantitative Data Value

Quantitative Data Element
Data Values

7/23/90
Page 1

Record Number: 1

Data Element Number: 47 Version Number: 1 Status ID: PR

Data Element Name:
Information Element Archival

Date

Data Value ID: QN

Rec Nbr:	Data Value Number	Data Value Definition Text:
1	Recent	Within one month.

2	late	Beyond required date.
---	------	-----------------------

D. Alias Element Options

Press any key to end the help

ALIAS MENU HELP

- LIST Aliases - Provides a view of a list of aliases and associated data element numbers that you may scroll through.
- ADD Aliases - Main Add form allows adding of Aliases, Associated data elements, and Qualitative or Quantitative Data Values.
- EDIT Aliases - Except for the "Alias Number," allows editing of all aspects of the Alias.
- DELETE Aliases- Allows deletion of Aliases. You must first delete all items under the associated [F3] menus (qual, quan, and host applic information).
- PRINT Reports - Moves to a lower Menu for a myriad of print options.
- HELP - This screen.

Press [F2] when finished viewing the table
Total records: 2

ALIAS-TO-DATA ELEMENT LIST FORM

Rec #	Alias #	Alias Name	Data Elmnt #
1	1	Accounting Code	1
2	2	Business Accounting Code	1

Press the ARROW keys to scroll through the values.


```

Comment
Text:

Source
List Text:

-----For Qualitative Data Values Only-----
Qualitative Data Value Accuracy Nbr %:
--Press [F3] to move in/out of the fields below. Use ARROW keys to scroll--
Rec#    Data Value Name                Data Value Definition Text

```

```

-----For Quantitative Data Values Only-----
Calc Fmla
Text:

Quantitative Data Value Accuracy ID:  [F1]           Quan Scale Nbr:
Quan Low Range Nbr:                   Quan High Range Nbr:

--Press [F3] to move in/out of the fields below.  Use ARROW keys to scroll--
Rec#  Data Value Number                Data Value Definition Text

                                           Press PgUp

```

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change

```

DATA ELEMENT ALIAS EDIT FORM
Rec#: 1

Where [F1] is shown, press the [F1] key for a list of choices.

Alias Number: 1 1 <
Alias Name
Accounting Code

Data Element Number: 1 [F1] Version Nbr: 1
Data Element Name
Information Reference Element Name

Data Value Type ID: QL
Max Length Characters: 34
PressPgDn

```

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change

```

Element Creator ID:
Timeliness Identifier: Qwe [F1]
Domain
Def Text:

Justification Cat: Left

—Press [F3] to move in/out of the fields below. Use ARROW keys to scroll—
Rec#   Host Applic Name   Host System Name   Int Fmt Cat   Dea Resp Ofc Nme

Press PgUp or PgDn

```

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change

Comment	
Text:	
Source	
List Text:	
-----For Qualitative Data Values Only-----	
Qualitative Data Value Accuracy Nbr %: 4	
-----Press [F3] to move in/out of the fields below. Use ARROW keys to scroll-----	
Rec#	Data Value Name Data Value Definition Text
1	Report Accounting code report.
2	Status Accounting Status.
-----Press PgUp or PgDn-----	

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change

-----For Quantitative Data Values Only-----	
Calc Fmla	
Text:	
Quantitative Data Value Accuracy ID: [F1] Quan Scale Nbr:	
Quan Low Range Nbr: Quan High Range Nbr:	
-----Press [F3] to move in/out of the fields below. Use ARROW keys to scroll-----	
Rec#	Data Value Number Data Value Definition Text
-----Press PgUp-----	

```
[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change
[Del] - Delete a record
```

DATA ELEMENT ALIAS DELETE FORM		Rec#: 1
To delete an Alias, first delete data in all fields listed under [F3] on pages 2-4. Then return to page 1 and press the [Del] key.		
Alias Number:	1	◀
Alias Name		
Accounting Code		
Data Element Number:	1	[F1]
Version Nbr:	1	
Data Element Name		
Information Reference Element	Name	
Data Value Type ID:	QL	Max Length Characters: 34
		PressPgDn

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change
[Del] - Delete a record

```

Element Creator ID:
Timeliness Identifier: Qwe [F1]
Domain
Def Text:
Justification Cat: Left

```

```

Press [F3] to move in/out of the fields below. Use ARROW keys to scroll
Rec#      Host Applic Name      Host System Name      Int Fmt Cat      Dea Resp Ofc Nme

```

Press PgUp or PgDn

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change
[Del] - Delete a record

Comment	
Text:	
Source	
List Text:	
—For Qualitative Data Values Only—	
Qualitative Data Value Accuracy Nbr #: 4	
—Press [F3] to move in/out of the fields below. Use ARROW keys to scroll—	
Rec#	Data Value Name Data Value Definition Text
1	Report Accounting code report.
2	Status Accounting Status.
—Press PgUp or PgDn	

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change
[Del] - Delete a record

—For Quantitative Data Values Only—	
Calc Fmla	
Text:	
Quantitative Data Value Accuracy ID: {F1} Quan Scale Nbr:	
Quan Low Range Nbr: Quan High Range Nbr:	
—Press [F3] to move in/out of the fields below. Use ARROW keys to scroll—	
Rec#	Data Value Number Data Value Definition Text
—Press PgUp	

Alias List Report

7/23/90
Page 1

Rec Nbr	Alias Nbr	Alias Name	Data Elmn Nbr
1	1	Accounting Code	1
2	2	Business Accounting Code	1

Data Element's Alias Name Report

7/23/90
Page 1

Record Number: 1

Data Element Number: 1

Data Element Name:
Information Reference Element
Name

Rec Nbr	Alias Nbr	Alias Name
1	1	Accounting Code

2	2	Business Accounting Code
---	---	--------------------------

Alias Report

7/23/90
Page 1

Record Number: 1

Alias Number: 1

Alias Name:
Accounting Code

Data Value Type ID: QL Max Length Characters: 34
Timeliness ID: Qwe Justification Category: Left
Creator ID:

Domain Definition Text:

Comment Text:

Source List Text:

Host Applic Name:

Host System Name:

Internal Format Category:
Alias Responsible Office Name:

If Qualitative, the following field applies:
Qualitative Data Value Accuracy Nbr %: 4

If Quantitative, the following fields apply:
Quantitative Accuracy ID: Quan Scale Number:
Quan Low Range Nbr: Quan High Range Nbr:

Rec Nbr	Data Elmnt Nbr	Data Elmnt Name
1	1	Information Reference Element

Name

Alias Qualitative Data Values

7/23/90
Page 1

Record Number: 1

Alias Number: 1

Alias Name:
Accounting Code

Rec Nbr	Data Value Name	Data Value Definition Text
1	Report	Accounting code report.
2	Status	Accounting Status.

Alias Quantitative Data Values

7/23/90
Page 1

Record Number: 1

Alias Number: 2

Alias Name:
Business Accounting Code

Rec Nbr	Data Value Number	Data Value Definition Text
-----	-----	-----
1	Business code dat	Business code date (accounting).
2	Business code val	Business code value (accounting).

E. Class Word Options

Press [F2] when finished viewing the table
Total records: 39

CLASS WORD LIST FORM			
Rec #	Class Word Name	Data Value Type ID	
1	Acceleration	QN	
2	Amount	QN	
3	Angle	QN	
4	Area	QN	
5	Category	OL	
6	Code	OL	
7	Cost	QN	
8	Date	QN	
9	Date-Time-Group	OL	
10	Density	QN	
11	Depth	QN	
12	Distance	QN	
13	Flow	QN	
14	Height	QN	

Use ARROW keys to scroll through the values.

[F2] - Data entry completed, Esc - Cancel data entry, Ctrl-U - Undo last change

CLASS WORD ADD FORM		
Class Word Name:	◀ Data Value Type ID:	Rec#: 1
Class Word Definition Text:		
Information Class Category Name:		

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change
[Del] - Delete a record

CLASS WORD EDIT and DELETE FORM		
Class Word Name: Acceleration	A=	Rec#: 1
Class Word Definition Text:	Data Value Type ID: QN	
rate of change of velocity.		
Information Class Category Name:		
<div style="border: 1px solid black; padding: 5px; margin: 10px auto; width: 80%;">In the "edit" mode, Class Word Name is "display" only. To delete, press the [Del] key.</div>		

Class Word List Report

7/23/90

Page 1

Rec Nbr	Class Word Name	Data Val Type	Class Word Definition Text
1	Acceleration	QN	The rate of change of velocity.
2	Amount	QN	A monetary value arrived at by counting, aggregation, or calculation.
3	Angle	QN	The space formed by two lines diverging from a common point.
4	Area	QN	The number of unit squares equal in measure to the surface of an object.
5	Category	QL	A division or subset in a system of classification in which all items share the same concept.
6	Code	QL	A designation for a specific object expressed in one or more characters. A set of qualitative, non-literal data, the specified internal structure of which is not easily determined without interpretation (decoding).
7	Cost	QN	The amount paid or required in payment for a purchase.
8	Date	QN	A notation of a specific 24 hour period of time expressed in the format year, month, and day; (YYYYMMDD).
9	Date-Time-Group	QL	A character string designating a specific date, time, and time zone in the format DDHHMMZMMYY (JCS).
10	Density	QN	The mass per unit volume of particular items of interest.

Class Word Report

7/23/90

Page 1

Record Number: 1

Class Word Name: Code

Data Value Type ID: QL

: Information Class Category Name:

Class Word Definition Text:

A designation for a specific object expressed in one or more characters.
A set of qualitative, non-literal data, the specified internal structure
of which is not easily determined without interpretation (decoding).

F. Prime Word Options

Press [F2] when finished viewing the table
Total records: 264

PRIME WORD LIST FORM		
Rec Nbr	Prime Word Name	Army Data Arch Subject-Area Name
1	Accounting	Budget
2	Acquisition	Acquisition
3	Administration	Support Activities
4	Affair	Public Affairs
5	Agency	
6	Agreement	Contracts
7	Air	Transportation
8	Air-Defence	Operations Plans
9	Air-Ground	Operations Plans
10	Aircraft	
11	Airfield	
12	Airlift	
13	Airport	

Use the ARROW keys to scroll through the values.

[F2] - Data entry completed, Esc - Cancel data entry, Ctrl-U - Undo last change

PRIME WORD FORM		Rec Nbr: 1
Prime Word Name:		
Army Data Architecture Subject-Area Name:		

[F2] - Complete edit, [Esc] - Cancel edit, Ctrl-U - Undo last change
[Del] - Delete a record

PRIME WORD EDIT/DELETE FORM		Rec Nbr:
Prime Word Name:	Accounting	1
Army Data Architecture Subject-Area Name:	Budget	

Press PgUp or PgDn

Prime Word List

7/23/90
Page 1

Prime Word Name	Army Data Architecture Subject-Area Name
Accounting	Budget
Acquisition	Acquisition
Administration	Support Activities
Affair	Public Affairs
Agency	
Agreement	Contracts
Air	Transportation
Air-Defence	Operations Plans
Air-Ground	Operations Plans
Aircraft	
Airfield	
Airlift	
Airport	
Alert	
Ammunition	
Anchorage	
Annex	Facilities
Appropriated	Funds
Apron	
Arctic	
Army	Unit(s) and Organization(s)
Arresting-Gear	
Arrival	
Assessment	Operations Plans
Asset	
Assistance	Security Assistance

APPENDIX H

A. USER'S MANUAL

1. Introduction

This manual will discuss in detail the operations of the prototype dictionary. The system forms and reports are displayed in Appendix F. The menu system is self-prompting, identifying computer keyboard options to run desired functions. Where operations become questionable, instructions will be more detailed.

2. General Operations

The following general operations are universal unless otherwise discussed:

1. Keyboard buttons in this manual are represented by placing brackets around the key symbol or word. For example, the "enter" key is [Enter] and the "A" key is [A].
2. Navigating menus. Because the menus are a horizontal bar-type, like Lotus 123, use the right and left arrow keys to move to a desired selection. Then press [Enter] to activate your selection.
3. Look-up tables. On several forms, certain fields, identified by a yellow [F1] on the right side of the data field, use look-up tables as a pool of options that can only be used for that field. Press [F1] to reveal the look-up table. A standard table will appear with data choices within it. You may use the [Arrows] to explore the table. Select the desired data option

by highlighting the correct horizontal line with your cursor and pressing [F2].

4. The [F2] key. In most form operations, pressing [F2] means "do-it." In other words, the selected operation will cease and all current data added, edited, viewed, or deleted data changes will be saved. The user will be returned to the menu.
5. Embedded forms can be identified by the [F3] key instruction heading (prior to the fields). The [F3] must be used to get into and out of embedded forms.

3. Initializing The System

At the "C:\>" prompt type "DODDICT" and press [Enter]. This Disk Operating System (DOS) batch file will initialize Paradox 3.0 and play the master script. When completed, the introduction/password screen will be shown. Enter the password "DODDICT" and press [Enter]. The main menu will replace the introduction/password screen.

4. The Reference Element Menu Selection

Upon making this selection, the user is presented with the following Reference Element options: (1) View a list of Reference Elements; (2) Add Reference Elements; (3) Edit a selected Reference Element; (4) Delete Reference Elements; (5) Move to a sub-menu of Reference Element printed report options; (6) Menu Help; (7) Leave and return to the next higher menu. Descriptions of these operations are as follows:

a. View a list of Reference Elements

This view displays record number, Reference Element number, Data Type (Qualitative or Quantitative), element status, and Reference Element name. This list can be scrolled by using the up and down [Arrows]. No update operations are permitted. Total number of records are posted at the top of the form. Press [F2] when done with the view.

b. Add Reference Elements

This four-page form (with two embedded fields) shows all of the attributes associated with Reference Element as well as options to make the Reference Element Qualitative or Quantitative. The cursor is first set on Reference Element Number. This is a required field; a number must be inserted, otherwise no more operations are permitted. If nothing is placed in the Version field, it will default to "1." Also, the Status ID will default to "PR," meaning Proposed (An [F1] option is available here). The Name is built with an optional Modifier, required Class Word, and 2 optional Qualifiers. As you finish each field, the complete name will appear in the narrow box below. An [F1] option exists with Class Word. If the Class Word field is bypassed, "REQUIRED FIELD" is inserted into both the Class Word field and the Name field. After a Class Word is

chosen, an associated Data Value type and Definition text is automatically inserted. The Data Value type is fixed based on the Class Word (Cannot be changed), but the Definition Text may be edited. The remaining attributes on the first and second page are self explanatory (Other [F1] options exist).

Page three is for Qualitative (QL) Reference Elements only. The Instructions are self-explanatory. Page four is for Quantitative (QN) Reference Elements only and is also self-explanatory. At the completion of page four, the operator has the option to [PgDn] to a new add form or end the add process.

To end the process the user may: (1) Press [F2] in order to save the new record and end the add process; (2) Press [Ctrl] and [U] together to delete his new record and start at a fresh record to continue the add process; (3) Press [Esc] to end the add process. Note, the user may have to place a number in the required first field in order to use the [Esc] option. This number will not be recorded as a new element.

c. Edit an Existing Reference Element

The user is asked for a Reference Element Number upon selection of this option. A four-page form, identical to the add form appears. The Reference Element Number field

cannot be edited. Introduction of a new number at this point would confuse the 1:N relationships in the embedded forms (The Qualitative and Quantitative options). Other than this, operations are identical in edit except for the fact that the user can only edit a Reference Element number that he first requests.

If new Qualitative Data Items or Quantitative Data Values need to be added or old ones subtracted, the edit form is the correct place to do it. Non-required attributes may be deleted in the edit mode. However, full Reference Element deletion cannot be done in this option.

d. Delete Reference Elements

The system does not ask for a number, but presents records in order. The user must use [PgDn] and [PgUp] to locate the desired record. The three-page form is different than the add and edit forms. To delete a Reference Element, the user must first delete all data in the embedded forms. Use [F3] to enter the embedded forms and then use the [Del] key. All links must be deleted prior to deleting the "master" record. This helps maintain data integrity by not permitting the Qualitative and Quantitative data to exist without a parent Reference Element. Once the embedded forms are cleared, the [Del] key can be pressed to remove the Reference Element.

e. Move to a Printed Report options sub-menu

This selection moves the user to a sub-menu with the following options: (1) Print a Reference Element list; (2) Print a detailed report on a Reference Element; (3) Print the Qualitative Data Items associated with a Reference Element; (4) Print the Quantitative Data Values associated with a Reference Element; (5) Leave: move to the next higher menu.

(1) *Print a Reference Element List.* This option routs an entire Reference Element list, similar to the view option, to the printer.

(2) *Print a detailed report on a Reference Element.* This option asks for a Reference Element number for input and then sends a detailed report with all of the Reference Element attributes, except Data Items and Data Values, to the printer.

(3) *Print the Qualitative Data Items associated with a Reference Element.* Based on the input of Reference Element number, this option prints all of the Data Items associated with a Reference Element. If the Reference Element does not have these values or is a Quantitative Data Value, "Nothing to report" will be flashed on the screen.

(4) *Print the Quantitative Data Values*
associated with a Reference Element. Based on the input of Reference Element number, this option prints all of the Data Values associated with a Reference Element. If the Reference Element does not have these values or is a Qualitative Data Value, "Nothing to report" will be flashed on the screen.

(5) *Leave.* This option lets you return to the next higher menu.

(6) *Menu Help.* This option describes each menu selection option in detail.

f. Leave and return to the next higher menu

This option is self-explanatory.

5. The Data Element Menu Selection

Upon making this selection, the user is presented with the following Data Element options: (1) View a list of Data Elements; (2) Add Data Elements; (3) Edit a selected Data Element; (4) Delete Data Elements; (5) Move to a sub-menu of Data Element printed report options; (6) Menu Help; (7) Leave and return to the next higher menu. Descriptions of these operations are as follows:

a. View a list of Data Elements

This view displays record number, Data Element number, Data Type (Qualitative or Quantitative), element

status, and Data Element name. This list can be scrolled by using the up and down [Arrows]. No update operations are permitted. Total number of records are posted at the top of the form. Press [F2] when done with the view.

b. Add Data Elements

This four-page form (with three embedded fields) shows all of the attributes associated with Data Element as well as options to make the Data Element Qualitative or Quantitative. The cursor is first set on Data Element Number. This is a required field; a number must be inserted, otherwise no more operations are permitted. If nothing is placed in the Version field, it will default to "1." Also, the Status ID will default to "PR," meaning Proposed (An [F1] option is available here). The Name is built with an optional six Modifiers and a required Prime Word. Note that twelve Modifiers are offered. This is because the Prime word may be positioned anywhere and this system must allow for six modifiers before or six after. Only choose six modifiers total. As you finish each field, the name will start appear in the narrow box below. An [F1] option exists with Prime Word. If the Prime Word field is bypassed, "REQUIRED FIELD" is inserted into both the Prime Word field and the Name field. Next, a Reference Element number is chosen as part of the Data Element. An [F1]

option is offered here. After the Reference Element is chosen, the associated Definition text is automatically inserted. This is because the domain of the Data Element must be the same or a subset of the Reference Element. The Data Value type is fixed based on the Reference Element (Cannot be changed), but the Definition Text may be edited. The remaining attributes on the first and second page are self explanatory (Other [F1] options exist).

Page three is for Qualitative (QL) Data Elements only. The Instructions are self-explanatory. Page four is for Quantitative (QN) Data Elements only and is also self-explanatory. At the completion of page four, the operator has the option to [PgDn] to a new add form or end the add process.

To end the process the user may: (1) Press [F2] in order to save the new record and end the add process; (2) Press [Ctrl] and [U] together to delete his new record and start at a fresh record to continue the add process; (3) Press [Esc] to end the add process. Note, the user may have to place a number in the required first field in ,order to use the [Esc] option. This number will not be recorded as a new element.

c. Edit an Existing Data Element

The user is asked for a Data Element Number upon selection of this option. A four-page form, identical to the add form appears. The Data Element Number field cannot be edited. Introduction of a new number at this point would confuse the 1:N relationships in the embedded forms (The Qualitative and Quantitative options). Other than this, operations are identical in edit except for the fact that the user can only edit a Data Element number that he first requests.

If new Qualitative Data Items or Quantitative Data Values need to be added or old ones subtracted, the edit form is the correct place to do it. Non-required attributes may be deleted in the edit mode. However, full Data Element deletion cannot be done in this option.

d. Delete Data Elements

The system does not ask for a number, but presents records in order. The user must use [PgDn] and [PgUp] to locate the desired record. The three-page form is different than the add and edit forms. To delete a Data Element, the user must first delete all data in the embedded forms. Use [F3] to enter the embedded forms and then use the [Del] key. All links must be deleted prior to deleting the "master" record. This helps maintain data integrity by

not permitting the Qualitative and Quantitative data to exist without a parent Data Element. Once the embedded forms are cleared, the [Del] key can be pressed to remove the Data Element.

e. *Move to a Printed Report options sub-menu*

This selection moves the user to a sub-menu with the following options: (1) Print a Data Element list; (2) Print a detailed report on a Data Element; (3) Print the Qualitative Data Items associated with a Data Element; (4) Print the Quantitative Data Values associated with a Data Element; (5) Leave: move to the next higher menu.

(1) *Print a Data Element List.* This option routes an entire Data Element list, similar to the view option, to the printer.

(2) *Print a detailed report on a Data Element.* This option asks for a Data Element number for input and then sends a detailed report with all of the Data Element attributes, except Data Items and Data Values, to the printer.

(3) *Print the Qualitative Data Items* associated with a Data Element. Based on the input of Data Element number, this option prints all of the Data Items associated with a Data Element. If the Data Element does

not have these values or is a Quantitative Data Value,
"Nothing to report" will be flashed on the screen.

(4) *Print the Quantitative Data Values*
associated with a Data Element. Based on the input of Data
Element number, this option prints all of the Data Values
associated with a Data Element. If the Data Element does
not have these values or is a Qualitative Data Value,
"Nothing to report" will be flashed on the screen.

(5) *Leave.* This option lets you return to the
next higher menu.

(6) *Menu Help.* This option describes each
menu selection option in detail.

f. Leave and return to the next higher menu

This option is self-explanatory.

6. The Alias Element Menu Selection

Upon making this selection, the user is presented
with the following Alias Element options: (1) View a list
of Alias Elements; (2) Add Alias Elements; (3) Edit a
selected Alias Element; (4) Delete Alias Elements; (5) Move
to a sub-menu of Alias Element printed report options; (6)
Menu Help; (7) Leave and return to the next higher menu.
Descriptions of these operations are as follows:

a. View a list of Alias Elements

This view displays record number, Alias Element number, Alias Element name, and associated Data Element numbers. This list can be scrolled by using the up and down [Arrows]. No update operations are permitted. Total number of records are posted at the top of the form. Press [F2] when done with the view.

b. Add Alias Elements

This four-page form (with three embedded fields) shows all of the attributes associated with Alias Element as well as options to make the Alias Element Qualitative or Quantitative. The cursor is first set on Alias Element Number. This is a required field; a number must be inserted, otherwise no more operations are permitted. Next the Alias Element name is inserted. Then, a Data Element must then be associated with the Alias. An [F1] option may be used here to select one associated Data Element. The remaining attributes on the first page and beginning of the second page are self explanatory (Other [F1] options exist). The latter half of the second page is an embedded form containing host system application data.

Page three is for Qualitative (QL) Alias Elements only. An embedded form is located on page three. The Instructions are self-explanatory. Page four is for

Quantitative (QN) Alias Elements only and is also self-explanatory. An embedded form is also located here. At the completion of page four, the operator has the option to [PgDn] to a new add form or end the add process.

To end the process the user may: (1) Press [F2] in order to save the new record and end the add process; (2) Press [Ctrl] and [U] together to delete his new record and start at a fresh record to continue the add process; (3) Press [Esc] to end the add process. Note, the user may have to place a number in the required first field in ,order to use the [Esc] option. This number will not be recorded as a new element.

c. Edit an Existing Alias Element

The user is asked for a Alias Element Number upon selection of this option. A four-page form, identical to the add form appears. The Alias Element Number field cannot be edited. Introduction of a new number at this point would confuse the 1:N relationships in the embedded forms (The Qualitative and Quantitative options). Other than this, operations are identical in edit except for the fact that the user can only edit an Alias Element number that he first requests.

If new Qualitative Data Items or Quantitative Data Values need to be added or old ones subtracted, the

edit form is the correct place to do it. Non-required attributes may be deleted in the edit mode. However, full Alias Element deletion cannot be done in this option.

d. Delete Alias Elements

The system does not ask for a number, but presents records in order. The user must use [PgDn] and [PgUp] to locate the desired record. The four-page form is slightly different than the add and edit forms. To delete a Alias Element, the user must first delete all data in the embedded forms. Use [F3] to enter the embedded forms and then use the [Del] key. All links must be deleted prior to deleting the "master" record. This helps maintain data integrity by not permitting the Qualitative and Quantitative data to exist without a parent Data Element. Once the embedded forms are cleared, the [Del] key can be pressed to remove the Alias Element.

e. Move to a Printed Report options sub-menu

This selection moves the user to a sub-menu with the following options: (1) Print a list of Alias Elements and their associated Data Element numbers; (2) From Data Element number input, print a list of associated Alias Elements; (3) Print a detailed report on a Alias Element; (4) Print the Qualitative Alias Items associated with a Alias Element; (5) Print the Quantitative Data Values

associated with a Alias Element; (6) Leave: move to the next higher menu.

(1) *Print an Alias Element List.* This option routs an entire Alias Element list with associated Data Element numbers, similar to the view option, to the printer.

(2) *From Data Element number input, print a list of associated Alias Elements.* Self-explanatory.

(3) *Print a detailed report on a Alias Element.* This option asks for a Alias Element number for input and then sends a detailed report with all of the Alias Element attributes, except Data Items and Data Values, to the printer.

(4) *Print the Qualitative Data Items associated with a Alias Element.* Based on the input of Alias Element number, this option prints all of the Data Items associated with a Alias Element. If the Alias Element does not have these values or is a Quantitative Data Value, "Nothing to report" will be flashed on the screen.

(5) *Print the Quantitative Data Values associated with an Alias Element.* Based on the input of Alias Element number, this option prints all of the Data Values associated with an Alias Element. If the Alias Element does not have these values or is a Qualitative Data Value, "Nothing to report" will be flashed on the screen.

(6) **Leave.** This option lets you return to the next higher menu.

(7) **Menu Help.** This option describes each menu selection option in detail.

f. Leave and return to the next higher menu

This option is self-explanatory.

7. The Class Word Menu Selection

Upon making this selection, the user is presented with the following Class Word options: (1) View a list of Class Words; (2) Add Class Words; (3) Edit/Delete Class Words; (4) Move to a sub-menu of Class Word printed report options; (5) Leave and return to the next higher menu.

Descriptions of these operations are as follows:

a. View a list of Class Words

This view displays record number, Class Word name, and Data Value Type ID. This list can be scrolled by using the up and down [Arrows]. No update operations are permitted. Total number of records are posted at the top of the form. Press [F2] when done with the view.

b. Add Class Words

This one-page form shows all of the attributes associated with a Class Word. The cursor is first set on Class Word name. This is a required field; a name must be

inserted, otherwise no more operations are permitted. The remaining attributes may then be added.

To end the process the user may: (1) Press [F2] in order to save the new record and end the add process; (2) Press [Ctrl] and [U] together to delete his new record and start at a fresh record to continue the add process; (3) Press [Esc] to end the add process. Note, the user may have to place a name in the required first field in order to use the [Esc] option. This name will not be recorded as a new element.

c. Edit/Delete an Existing Class Word

The system does not ask for a name, but presents the records in order. Class Words are a finite set. A one-page form appears. All fields can be edited. To delete a Class Word, you must first delete associated Reference Elements. Then press the [Del] key.

d. Move to a Printed Report options sub-menu

This selection moves the user to a sub-menu with the following options: (1) Print a list of Class Words; (2) Print a detailed report on a Class Word.

(1) *Print a Class Word List.* This option prints an entire Class Word list with associated Data Value type ID's.

(2) From Class Word name input, print a detailed report on a Class Word. This option is self-explanatory.

(3) Leave. This option lets you return to the next higher menu.

e. Leave and return to the next higher menu

This option is self-explanatory.

8. The Prime Word Menu Selection

Upon making this selection, the user is presented with the following Prime Word options: (1) View a list of Prime Words; (2) Add Prime Words; (3) Edit/Delete Prime Words; (4) Move to a sub-menu of Class Word printed report options; (5) Leave and return to the next higher menu.

Descriptions of these operations are as follows:

a. View a list of Prime Words

This view displays record number, Prime Word name, and Data Architecture Subject-area Name. This list can be scrolled by using the up and down [Arrows]. No update operations are permitted. Total number of records are posted at the top of the form. Press [F2] when done with the view.

b. Add Prime Words

This one-page form shows all of the attributes associated with a Prime Word. The cursor is first set on

Prime Word name. This is a required field; a name must be inserted, otherwise no more operations are permitted. The remaining attribute Army Data Architecture Subject-area Name, may then be added.

To end the process the user may: (1) Press [F2] in order to save the new record and end the add process; (2) Press [Ctrl] and [U] together to delete his new record and start at a fresh record to continue the add process; (3) Press [Esc] to end the add process. Note, the user may have to place a name in the required first field in ,order to use the [Esc] option. This name will not be recorded as a new element.

c. Edit/Delete an Existing Prime Word

The system does not ask for a name, but presents the records in order. Prime Words are a finite set. A one-page form appears. All fields can be edited. To delete a Prime Word, you must first delete associated Data Elements. Then press the [Del] key.

d. Print a List of Prime Words

This option sends a report listing the Prime Words and Data Subject-areas to the printer.

e. Leave and return to the next higher menu

This option is self-explanatory.

9. The Unique Query Menu Option

This option is set-aside for future development based on user needs.

10. The Main Menu Help Option

Selection of this option displays a one-page help screen which diagrams the menu hierarchy and gives brief instruction on how to navigate through the menu hierarchy structure.

11. The Leave Application Menu Option

This option lets you leave the application.

APPENDIX I

```
;*****
; Dname Script
; Paradox 3.0 PAL
; Master Application Script. Calls other sub-scripts.
; Jack Bacheller 21Jul90
;*****

if (sysmode() <> "Main") then
    Message "The application can only be started from Paradox
main mode"
    Sleep 3000
    return
endif

Echo Off
Clear
Reset
Cursor Off

; put up the greeting screen
@ 2, 0
Play "Dnameg"

; ask for the password to the application; this password
determines
; the access to the tables in the application allowed for
the
; current user of the application.

@ 0, 0
Style Attribute SysColor(0)
?? fill(" ",160)
@ 1, 0
?? "Enter password for the application; [Esc] to cancel;
[Enter] for no password."
@ 0, 0
?? "Password: "
Cursor Normal
zzzcolor = int(SysColor(0) / 16)
```

```

Style Attribute ((zzzcolor * 16) + zzzcolor)
Accept "a50" To pword
Style
EscEnter = not retval
Cursor Off

if (EscEnter) then
    Message "Cancelling the application"
    Sleep 2000
    Clear
    return
endif

if (pword <> "") then
    Password pword
endif

; set up the error proc for the application

ReadLib "Dnameut1" ApplicErrorProc
ErrorProc = "ApplicErrorProc"
ApplicErrorRetVal = FALSE

; Start the application

ReadLib "Dname1" Dname1Menu
Dname1Menu()
Release Procs Dname1Menu

Clearall
if (pword <> "") then
    UnPassword pword
endif

```

```
;*****
; Dname1 Script
; Paradox 3.0 PAL
; Provides the logic for the main menu, calls other scripts.
; Jack Bacheller 21Jul90
;*****
```

```
AppLib = "Dname1"
if (not isfile(AppLib + ".lib")) then
    Createlib AppLib
endif
```

```
proc Dname1Menu()
private x, escape, zzzmexit, zzzzexit, pword
```

```
    zzzzexit = FALSE
    x = "Reference Elements"
    while (TRUE)
        Clear
```

```
        ShowMenu
            "Reference Elements": "View, add, edit, delete,
print Reference Elements.",
            "Data Elements": "View, add, edit, delete, and print
Data Elements.",
            "Aliases": "View, add, edit, delete, and print Data
Element Aliases.",
            "Class Words": "View, add, edit, delete, print Class
Words.",
            "Prime Words": "View, add, edit, delete, and print
Prime Words.",
            "Unique Queries": "View special queries of
Information classes, subject areas.",
            "HELP": "Help screen describing dictionary purpose
and layout.",
            "Leave": "Leave the application"
        Default x
        To x
```

```
    switch
        case x = "Reference Elements":
            Play "Ref"
            x = "Reference Elements"
            escape = FALSE
```

```
        case x = "Data Elements":
            Play "Dat"
```

```

    x = "Data Elements"
    escape = FALSE

case x = "Aliases":
    Play "Dea"
    x = "Aliases"
    escape = FALSE

case x = "Class Words":
    Play "Cwd"
    x = "Class Words"
    escape = FALSE

case x = "Prime Words":
    Play "Pwd"
    x = "Prime Words"
    escape = FALSE

case x = "Unique Queries":
    ReadLib "Dnameut1" PlayHelp

PlayHelp("Dnameh2")
Release Procs PlayHelp

    escape = FALSE

case x = "HELP":
    ReadLib "Dnameut1" PlayHelp

PlayHelp("Dnameh1")
Release Procs PlayHelp

    escape = FALSE

case x = "Leave":
    ShowMenu
        "No": "Do not leave the application.",
        "Yes": "Leave the application."
    To zzzmexit

    zzzzexit = (zzzmexit = "Yes")
    escape = (zzzmexit = "Esc")

case x = "Esc":
    escape = FALSE
endswitch

Reset

```

```

; reset ErrorProc value
ErrorProc = "ApplicErrorProc"
ApplicErrorRetVal = FALSE

if (zzzzexit) then
    return TRUE
endif

if (not escape) then
    x = "Reference Elements"
endif
endwhile
endproc

Writelib AppLib Dname1Menu
Release Procs Dname1Menu

```

```
;*****  
; Ref Script  
; Paradox 3.0 PAL  
; Reference Element Main Script. Calls other Ref Elmnt  
scripts.  
; Jack Bacheller 21Jul90  
;*****
```

```
Echo Off  
Clear  
Reset  
Cursor Off
```

```
; set up the error proc for the application
```

```
ReadLib "Refutl" ApplicErrorProc  
ErrorProc = "ApplicErrorProc"  
ApplicErrorRetVal = FALSE
```

```
; Start the application
```

```
ReadLib "Ref1" Ref1Menu  
Ref1Menu()  
Release Procs Ref1Menu
```

```
Clearall
```

```
;*****
; Ref1 menu script
; Paradox 3.0 PAL
; Reference Element main script menu operations.
; Jack Bacheller 21Jul90
;*****
```

```
AppLib = "Ref1"
if (not isfile(AppLib + ".lib")) then
    Createlib AppLib
endif
```

```
proc Ref1S1()
private opResult
```

```
    Readlib "Refut1" ViewTable, ToggleForm, VwFldView,
        HelpKey
```

```
    opResult = ViewTable("Apwtrelm", "Apwtrelm", "5", FALSE)
```

```
    Release Procs ViewTable, ToggleForm, VwFldView,
        HelpKey
```

```
    return opResult
endproc
```

```
Writelib AppLib Ref1S1
Release Procs Ref1S1
```

```
proc Ref1S2()
private opResult
```

```
    Readlib "Refut1" EntryTable, KECheck, ToggleForm,
        EdFldView, HelpKey, EntryCancel, EntryDoIt,
        RenamePre, RenameSet, SaveList, CreateList,
        PrintList
```

```
    opResult = EntryTable("Apwtrelm", "", "7", FALSE)
```

```
    Release Procs EntryTable, KECheck, ToggleForm,
        EdFldView, HelpKey, EntryCancel, EntryDoIt,
        RenamePre, RenameSet, SaveList, CreateList,
        PrintList
```

```
    return opResult
endproc
```

```
Writelib AppLib Ref1S2
Release Procs Ref1S2
```

```
proc Ref1S3()
private opResult, tbl, rt, EscEnter, count
```

```
    Play "Refq1"      ; put query on workspace
    if (ApplicErrorRetVal) then
        ClearAll
        return FALSE
    endif
```

```
    Readlib "Refut1" EnterVal
; get value for variable count
    count = EnterVal("Enter the Reference Element Number
that you wish to edit. ", "N", "", 1)
    Release Procs EnterVal
    if (EscEnter) then
        ClearAll
        return FALSE
    endif
```

```
    Readlib "Refut1" QueryDoIt
    rt = QueryDoIt()
    Release Procs QueryDoIt
```

```
    if (not rt) then
        return FALSE
    endif
```

```
    if (isempty("Answer")) then
        Message "No records to edit"
        Sleep 3000
        return FALSE
    endif
```

```
    if (ApplicErrorRetVal) then
        return FALSE
    endif
```

```
    Readlib "Refut1" EditTable, ToggleForm, EdFldView,
        HelpKey, EditCancel, QEditDoIt
```

```
    opResult = EditTable("Answer", "Apwtrelm", "", "8", FALSE,
        "QEditDoIt", "",
        "",
        TRUE, FALSE, TRUE)
```

```

        Release Procs EditTable, ToggleForm, EdFldView,
            HelpKey, EditCancel, QEditDoIt

        return opResult
    endproc

Writelib AppLib Ref1S3
Release Procs Ref1S3

proc Ref1S4()
private opResult

    if (isempty("Apwtrelm")) then
        Message "No records to edit"
        Sleep 3000
        return FALSE
    endif

    if (ApplicErrorRetVal) then
        return FALSE
    endif

    Readlib "Refut1" EditTable, ToggleForm, EdFldView,
        HelpKey, EditCancel, SEditDoIt, SEditDelNoIns

    opResult = EditTable("Apwtrelm", "Apwtrelm", "", "6",
FALSE,
        "SEditDoIt", "SEditDelNoIns",
        "[Del] - Delete a record",
        TRUE, FALSE, FALSE)

    Release Procs EditTable, ToggleForm, EdFldView,
        HelpKey, EditCancel, SEditDoIt, SEditDelNoIns

    return opResult
endproc

Writelib AppLib Ref1S4
Release Procs Ref1S4

proc ReflMenu()
private x, escape, zzzmexit, zzzzexit

    zzzzexit = FALSE
    x = "LIST Reference Elmts"
    while (TRUE)

```

Clear

ShowMenu

```
"LIST Reference Elmts": "View a listing of Reference  
Elements (#,Name,Type Status).",  
"ADD Reference Elmts": "Add Qualitative or  
Quantitative Reference Elements.",  
"EDIT Reference Elmts": "Edit selected Qualitative  
or Quantitative Reference Elements",  
"DELETE Reference Elm": "Delete Qualitative or  
Quantitative Reference Elements.",  
"PRINT REPORTS": "Move to the PRINT REPORTS menu for  
Report selection.",  
"HELP": "Lists Help for this menu level.",  
"Leave": "Leave the Reference Element Application."
```

Default x

To x

switch

```
case x = "LIST Reference Elmts":
```

```
  ReadLib "Ref1" Ref1S1  
  escape = Ref1S1()  
  escape = not escape  
  Release Procs Ref1S1
```

```
case x = "ADD Reference Elmts":
```

```
  ReadLib "Ref1" Ref1S2  
  escape = Ref1S2()  
  escape = not escape  
  Release Procs Ref1S2
```

```
case x = "EDIT Reference Elmts":
```

```
  ReadLib "Ref1" Ref1S3  
  escape = Ref1S3()  
  escape = not escape  
  Release Procs Ref1S3
```

```
case x = "DELETE Reference Elm":
```

```
  ReadLib "Ref1" Ref1S4  
  escape = Ref1S4()  
  escape = not escape  
  Release Procs Ref1S4
```

```
case x = "PRINT REPORTS":
```

```
  Play "Refpt"  
  x = "PRINT REPORTS"  
  escape = FALSE
```

```

        case x = "HELP":
            ReadLib "Refut1" PlayHelp

PlayHelp("Refh1")
Release Procs PlayHelp

        escape = FALSE

        case x = "Leave":
            ShowMenu
                "No": "Do not leave the Reference Element
application.",
                "Yes": "Leave the Reference Element
application."
            To zzzmexit

            zzzzexit = (zzzmexit = "Yes")
            escape = (zzzmexit = "Esc")

        case x = "Esc":
            escape = FALSE
endswitch

Reset
; reset ErrorProc value
ErrorProc = "ApplicErrorProc"
ApplicErrorRetVal = FALSE

if (zzzzexit) then
    return TRUE
endif

if (not escape) then
    x = "LIST Reference Elmts"
endif
endwhile
endproc

Writelib AppLib ReflMenu
Release Procs ReflMenu

```

```
;*****
; Refpt script
; Paradox 3.0 PAL
; Reference Element Print options main script.
; Jack Bacheller 21Jul90
;*****
```

```
Echo Off
Clear
Reset
Cursor Off
```

```
; set up the error proc for the application
```

```
ReadLib "Refptut1" ApplicErrorProc
ErrorProc = "ApplicErrorProc"
ApplicErrorRetVal = FALSE
```

```
; Start the application
```

```
ReadLib "Refpt1" Refpt1Menu
Refpt1Menu()
Release Procs Refpt1Menu
```

```
Clearall
```

```

;*****
; Refpt1 Script
; Paraadox 3.0 PAL
; Reference Element Print operations.
; Jack Bacheller 21Jul90
;*****

AppLib = "Refpt1"
if (not isfile(AppLib + ".lib")) then
    Createlib AppLib
endif

proc Refpt1S1()
private opResult

    Readlib "Refptut1" ReportTable

    opResult = ReportTable("Apwtrelm", "Apwtrelm", "R",
        "Printer", "")

    Release Procs ReportTable

    return opResult
endproc

Writelib AppLib Refpt1S1
Release Procs Refpt1S1

proc Refpt1S2()
private opResult, tbl, rt, EscEnter, coung

    Play "Refptq3" ; put query on workspace
    if (ApplicErrorRetVal) then
        ClearAll
        return FALSE
    endif

    Readlib "Refptut1" EnterVal
; get value for variable coung
    coung = EnterVal("Enter the Reference Element Number.
", "N", "", 0)
    Release Procs EnterVal
    if (EscEnter) then
        ClearAll
        return FALSE
    endif

```

```

Readlib "Refptut1" QueryDoIt
rt = QueryDoIt()
Release Procs QueryDoIt

if (not rt) then
    return FALSE
endif

Readlib "Refptut1" ReportTable

opResult = ReportTable("Answer", "Apwtrelm", "1",
"Printer", "")

Release Procs ReportTable

return opResult
endproc

Writelib AppLib Refpt1S2
Release Procs Refpt1S2

proc Refpt1S3()
private opResult, tbl, rt, EscEnter, counh

    Play "Refptq1" ; put query on workspace
    if (ApplicErrorRetVal) then
        ClearAll
        return FALSE
    endif

    Readlib "Refptut1" EnterVal
; get value for variable counh
    counh = EnterVal("Enter the Reference Element Number.
", "N", "", 0)
    Release Procs EnterVal
    if (EscEnter) then
        ClearAll
        return FALSE
    endif

    Readlib "Refptut1" QueryDoIt
    rt = QueryDoIt()
    Release Procs QueryDoIt

    if (not rt) then
        return FALSE
    endif

```

```

Readlib "Refptut1" ReportTable

opResult = ReportTable("Answer", "Apwtrefd", "R",
"Printer", "")

Release Procs ReportTable

return opResult
endproc

Writelib AppLib Refpt1S3
Release Procs Refpt1S3

proc Refpt1S4()
private opResult, tbl, rt, EscEnter, counj

Play "Refptq2" ; put query on workspace
if (ApplicErrorRetVal) then
  ClearAll
  return FALSE
endif

Readlib "Refptut1" EnterVal
; get value for variable counj
counj = EnterVal("Enter the Reference Element Number.
", "N", "", 0)
Release Procs EnterVal
if (EscEnter) then
  ClearAll
  return FALSE
endif

Readlib "Refptut1" QueryDoIt
rt = QueryDoIt()
Release Procs QueryDoIt

if (not rt) then
  return FALSE
endif

Readlib "Refptut1" ReportTable

opResult = ReportTable("Answer", "Apwtreqd", "R",
"Printer", "")

Release Procs ReportTable

```

```

    return opResult
endproc

Writelib AppLib Refpt1S4
Release Procs Refpt1S4

proc Refpt1Menu()
private x, escape, zzzmexit, zzzzexit

    zzzzexit = FALSE
    x = "PRINT Ref Elmt List"
    while (TRUE)
        Clear

        ShowMenu
            "PRINT Ref Elmt List": "Print a combined list of
Reference Elmts (#,Type,Name,Stat).",
            "PRINT Ref Element": "Print a report on a selected,
specific Reference Element.",
            "PRINT Qual Data Vals": "Print the Qualitative Data
Values for a selected Ref Elmnt.",
            "PRINT Quan Data Vals": "Print the Data Values for a
selected Quantitative Ref Elmnt.",
            "Leave": "Leave the application"
        Default x
        To x

    switch
        case x = "PRINT Ref Elmt List":
            ReadLib "Refpt1" Refpt1S1
            escape = Refpt1S1()
            escape = not escape
            Release Procs Refpt1S1

        case x = "PRINT Ref Element":
            ReadLib "Refpt1" Refpt1S2
            escape = Refpt1S2()
            escape = not escape
            Release Procs Refpt1S2

        case x = "PRINT Qual Data Vals":
            ReadLib "Refpt1" Refpt1S3
            escape = Refpt1S3()
            escape = not escape
            Release Procs Refpt1S3

        case x = "PRINT Quan Data Vals":
            ReadLib "Refpt1" Refpt1S4

```

```

        escape = Refpt1S4()
        escape = not escape
        Release Procs Refpt1S4

    case x = "Leave":
        ShowMenu
            "No": "Do not leave the application.",
            "Yes": "Leave the application."
        To zzzmexit

        zzzzexit = (zzzmexit = "Yes")
        escape = (zzzmexit = "Esc")

    case x = "Esc":
        escape = FALSE
endswitch

Reset
; reset ErrorProc value
ErrorProc = "ApplicErrorProc"
ApplicErrorRetVal = FALSE

if (zzzzexit) then
    return TRUE
endif

if (not escape) then
    x = "PRINT Ref Elmt List"
endif
endwhile
endproc

Writelib AppLib Refpt1Menu
Release Procs Refpt1Menu

```

```

;*****
; Dat Script
; Paradox 3.0 PAL
; Data Element main script
; Jack Bacheller 21Jul90
;*****

Echo Off
Clear
Reset
Cursor Off

; set up the error proc for the application

ReadLib "Datut1" ApplicErrorProc
ErrorProc = "ApplicErrorProc"
ApplicErrorRetVal = FALSE

; Start the application

ReadLib "Dat1" Dat1Menu
Dat1Menu()
Release Procs Dat1Menu

Clearall

```

```
;*****
; Dat1 Script
; Paradox 3.0 PAL
; Data Element main menu operations.
; Jack Bacheller 21Jul90
;*****
```

```
AppLib = "Dat1"
if (not isfile(AppLib + ".lib")) then
  Createlib AppLib
endif
```

```
proc Dat1S1()
private opResult
```

```
  Readlib "Datut1" ViewTable, ToggleForm, VwFldView,
    HelpKey
```

```
  opResult = ViewTable("Apwtde", "Apwtde", "4", FALSE)
```

```
  Release Procs ViewTable, ToggleForm, VwFldView,
    HelpKey
```

```
  return opResult
endproc
```

```
Writelib AppLib Dat1S1
Release Procs Dat1S1
```

```
proc Dat1S2()
private opResult
```

```
  Readlib "Datut1" EntryTable, KECheck, ToggleForm,
    EdFldView, HelpKey, EntryCancel, EntryDoIt,
    RenamePre, RenameSet, SaveList, CreateList,
    PrintList
```

```
  opResult = EntryTable("Apwtde", "", "8", FALSE)
```

```
  Release Procs EntryTable, KECheck, ToggleForm,
    EdFldView, HelpKey, EntryCancel, EntryDoIt,
    RenamePre, RenameSet, SaveList, CreateList,
    PrintList
```

```
  return opResult
endproc
```

```
Writelib AppLib Dat1S2
Release Procs Dat1S2
```

```
proc Dat1S3()
private opResult, tbl, rt, EscEnter, coub

  Play "Datq2"      ; put query on workspace
  if (ApplicErrorRetVal) then
    ClearAll
    return FALSE
  endif

  Readlib "Datut1" EnterVal
; get value for variable coub
    coub = EnterVal("Enter the Data Element Number. ",
"N", "", 0)
  Release Procs EnterVal
  if (EscEnter) then
    ClearAll
    return FALSE
  endif

  Readlib "Datut1" QueryDoIt
  rt = QueryDoIt()
  Release Procs QueryDoIt

  if (not rt) then
    return FALSE
  endif

  if (isempty("Answer")) then
    Message "No records to edit"
    Sleep 3000
    return FALSE
  endif

  if (ApplicErrorRetVal) then
    return FALSE
  endif

  Readlib "Datut1" EditTable, ToggleForm, EdFldView,
    HelpKey, EditCancel, QEditDoIt

  opResult = EditTable("Answer", "Apwtde", "", "9", FALSE,
    "QEditDoIt", "",
    "",
    TRUE, FALSE, TRUE)
```

```

Release Procs EditTable, ToggleForm, EdFldView,
      HelpKey, EditCancel, QEditDoIt

      return opResult
endproc

Writelib AppLib Dat1S3
Release Procs Dat1S3

proc Dat1S4()
private opResult

      if (isempty("Apwtde")) then
            Message "No records to edit"
            Sleep 3000
            return FALSE
      endif

      if (ApplicErrorRetVal) then
            return FALSE
      endif

      Readlib "Datut1" EditTable, ToggleForm, EdFldView,
            HelpKey, EditCancel, SEditDoIt, SEditDelNoIns

      opResult = EditTable("Apwtde", "Apwtde", "", "10", FALSE,
            "SEditDoIt", "SEditDelNoIns",
            "[Del] - Delete a record",
            TRUE, FALSE, FALSE)

      Release Procs EditTable, ToggleForm, EdFldView,
            HelpKey, EditCancel, SEditDoIt, SEditDelNoIns

      return opResult
endproc

Writelib AppLib Dat1S4
Release Procs Dat1S4

proc Dat1Menu()
private x, escape, zzzmexit, zzzzexit

      zzzzexit = FALSE
      x = "LIST Data Elements"
      while (TRUE)
            Clear

```

```

ShowMenu
  "LIST Data Elements": "View a list of Qualitative
and Quantitative Data Elements.",
  "ADD Data Elements": "Add Qualitative or
Quantitative Data Elements.",
  "EDIT Data Elements": "Edit a specific, selected
Qualitative or Quantitative Dat El",
  "DELETE Data Elements": "Delete Qualitative or
Quantitative Data Elements.",
  "PRINT Data Elements": "Move to a lower menu for
Data Element print options.",
  "HELP": "Help on this menu.",
  "Leave": "Leave the application"
Default x
To x

```

```

switch
case x = "LIST Data Elements":
  ReadLib "Dat1" Dat1S1
  escape = Dat1S1()
  escape = not escape
  Release Procs Dat1S1

case x = "ADD Data Elements":
  ReadLib "Dat1" Dat1S2
  escape = Dat1S2()
  escape = not escape
  Release Procs Dat1S2

case x = "EDIT Data Elements":
  ReadLib "Dat1" Dat1S3
  escape = Dat1S3()
  escape = not escape
  Release Procs Dat1S3

case x = "DELETE Data Elements":
  ReadLib "Dat1" Dat1S4
  escape = Dat1S4()
  escape = not escape
  Release Procs Dat1S4

case x = "PRINT Data Elements":
  Play "Datpt"
  x = "PRINT Data Elements"
  escape = FALSE

case x = "HELP":
  ReadLib "Datut1" PlayHelp

```

```

PlayHelp("Dath1")
Release Procs PlayHelp

    escape = FALSE

    case x = "Leave":
        ShowMenu
            "No": "Do not leave the application.",
            "Yes": "Leave the application."
        To zzzmexit

        zzzzexit = (zzzmexit = "Yes")
        escape = (zzzmexit = "Esc")

    case x = "Esc":
        escape = FALSE
    endswitch

Reset
; reset ErrorProc value
ErrorProc = "ApplicErrorProc"
ApplicErrorRetVal = FALSE

if (zzzzexit) then
    return TRUE
endif

if (not escape) then
    x = "LIST Data Elements"
endif
endwhile
endproc

Writelib AppLib Dat1Menu
Release Procs Dat1Menu

```

```

;*****
; Datpt Script
; Paradox 3.0 PAL
; Data Element Print options main script
; Jack Bacheller 21Jul90
;*****

```

```

Echo Off
Clear
Reset
Cursor Off

```

```

; set up the error proc for the application

```

```

ReadLib "Datptut1" ApplicErrorProc
ErrorProc = "ApplicErrorProc"
ApplicErrorRetVal = FALSE

```

```

; Start the application

```

```

ReadLib "Datpt1" Datpt1Menu
Datpt1Menu()
Release Procs Datpt1Menu

```

```

Clearall

```

```

;*****
; Datpt1 Script
; Paradox 3.0 PAL
; Data Element Print operations
; Jack Bacheller
;*****

```

```

AppLib = "Datpt1"
if (not isfile(AppLib + ".lib")) then
    Createlib AppLib
endif

```

```

proc Datpt1S1()

```

```

private opResult

  Readlib "Datptutl" ReportTable

  opResult = ReportTable("Apwtd", "Apwtd", "1", "Printer",
  "")

  Release Procs ReportTable

  return opResult
endproc

Writelib AppLib Datpt1S1
Release Procs Datpt1S1

proc Datpt1S2()
private opResult, tbl, rt, EscEnter, couh

  Play "Datptq1" ; put query on workspace
  if (ApplicErrorRetVal) then
    ClearAll
    return FALSE
  endif

  Readlib "Datptutl" EnterVal
; get value for variable couh
  couh = EnterVal("Enter the Data Element Number: ",
  "N", "", 0)
  Release Procs EnterVal
  if (EscEnter) then
    ClearAll
    return FALSE
  endif

  Readlib "Datptutl" QueryDoIt
  rt = QueryDoIt()
  Release Procs QueryDoIt

  if (not rt) then
    return FALSE
  endif

  Readlib "Datptutl" ReportTable

  opResult = ReportTable("Answer", "Apwtd", "R", "Printer",
  "")

```

```

Release Procs ReportTable

    return opResult
endproc

Writelib AppLib Datpt1S2
Release Procs Datpt1S2

proc Datpt1S3()
private opResult, tbl, rt, EscEnter, couw

    Play "Datptq2"      ; put query on workspace
    if (ApplicErrorRetVal) then
        ClearAll
        return FALSE
    endif

    Readlib "Datptut1" EnterVal
; get value for variable couw
        couw := EnterVal("Enter the Data Element Number: ",
"N", "", 0)
    Release Procs EnterVal
    if (EscEnter) then
        ClearAll
        return FALSE
    endif

    Readlib "Datptut1" QueryDoIt
    rt = QueryDoIt()
    Release Procs QueryDoIt

    if (not rt) then
        return FALSE
    endif

    Readlib "Datptut1" ReportTable

    opResult = ReportTable("Answer", "Apwtdedi", "R",
"Printer", "")

    Release Procs ReportTable

    return opResult
endproc

Writelib AppLib Datpt1S3
Release Procs Datpt1S3

```

```

proc Datpt1S4()
private opResult, tbl, rt, EscEnter, couq

    Play "Datptq3"      ; put query on workspace
    if (ApplicErrorRetVal) then
        ClearAll
        return FALSE
    endif

    Readlib "Datptutl" EnterVal
; get value for variable couq
        couq = EnterVal("Enter the Data Element Number: ",
"N", "", 0)
    Release Procs EnterVal
    if (EscEnter) then
        ClearAll
        return FALSE
    endif

    Readlib "Datptutl" QueryDoIt
    rt = QueryDoIt()
    Release Procs QueryDoIt

    if (not rt) then
        return FALSE
    endif

    Readlib "Datptutl" ReportTable

    opResult = ReportTable("Answer", "Apwtdeqd", "R",
"Printer", "")

    Release Procs ReportTable

    return opResult
endproc

Writelib AppLib Datpt1S4
Release Procs Datpt1S4

proc Datpt1Menu()
private x, escape, zzzmexit, zzzzexit

    zzzzexit = FALSE
    x = "PRINT Data Elmt List"
    while (TRUE)
        Clear

```

```

ShowMenu
  "PRINT Data Elmt List": "Print a list of Qualitative
and Quantitative Data Elements.",
  "PRINT Data Element": "Print a detailed report on a
specific, selected Data Element",
  "PRINT Qual Data Vals": "Print the Qualitative Data
Values of a selected Data Element",
  "PRINT Quan Data Vals": "Print the Quantitative Data
Values of a selected Data Elmnt.",
  "Leave": "Leave the application"
Default x
To x

```

```

switch
case x = "PRINT Data Elmt List":
  ReadLib "Datpt1" Datpt1S1
  escape = Datpt1S1()
  escape = not escape
  Release Procs Datpt1S1

case x = "PRINT Data Element":
  ReadLib "Datpt1" Datpt1S2
  escape = Datpt1S2()
  escape = not escape
  Release Procs Datpt1S2

case x = "PRINT Qual Data Vals":
  ReadLib "Datpt1" Datpt1S3
  escape = Datpt1S3()
  escape = not escape
  Release Procs Datpt1S3

case x = "PRINT Quan Data Vals":
  ReadLib "Datpt1" Datpt1S4
  escape = Datpt1S4()
  escape = not escape
  Release Procs Datpt1S4

case x = "Leave":
  ShowMenu
    "No": "Do not leave the application.",
    "Yes": "Leave the application."
  To zzzmexit

  zzzzexit = (zzzmexit = "Yes")
  escape = (zzzmexit = "Esc")

case x = "Esc":

```

```

        escape = FALSE
    endswitch

    Reset
    ; reset ErrorProc value
    ErrorProc = "ApplicErrorProc"
    ApplicErrorRetVal = FALSE

    if (zzzzexit) then
        return TRUE
    endif

    if (not escape) then
        x = "PRINT Data Elmt List"
    endif
endwhile
endproc

Writelib AppLib DatptlMenu
Release Procs DatptlMenu

```

```

;*****
; Dea Script
; Paradox 3.0 PAL
; Data Element Alias Main Script
; Jack Bacheller 21Jul90
;*****

Echo Off
Clear
Reset
Cursor Off

; set up the error proc for the application

ReadLib "Deautl" ApplicErrorProc
ErrorProc = "ApplicErrorProc"
ApplicErrorRetVal = FALSE

; Start the application

ReadLib "Deal" DealMenu
DealMenu()
Release Procs DealMenu

Clearall

```

```

;*****
; Deal Script
; Paradox 3.0 PAL
; Data Element Alias Main menu operations.
; Jack Bacheller 21Jul90
;*****

```

```

AppLib = "Deal"
if (not isfile(AppLib + ".lib")) then
  Createlib AppLib
endif

```

```

proc DealS1()
private opResult

  Readlib "Deautl" ViewTable, ToggleForm, VwFldView,
    HelpKey

  opResult = ViewTable("Apwtal", "Apwtal", "3", FALSE)

  Release Procs ViewTable, ToggleForm, VwFldView,
    HelpKey

  return opResult
endproc

```

```

Writelib AppLib DealS1
Release Procs DealS1

```

```

proc DealS2()
private opResult

  Readlib "Deautl" EntryTable, KECheck, ToggleForm,
    EdFldView, HelpKey, EntryCancel, EntryDoIt,
    RenamePre, RenameSet, SaveList, CreateList,
    PrintList

  opResult = EntryTable("Apwtal", "", "F", FALSE)

  Release Procs EntryTable, KECheck, ToggleForm,
    EdFldView, HelpKey, EntryCancel, EntryDoIt,
    RenamePre, RenameSet, SaveList, CreateList,
    PrintList

  return opResult

```

endproc

Writelib AppLib DealS2
Release Procs DealS2

proc DealS3()
private opResult

if (isempty("Apwtal")) then
 Message "No records to edit"
 Sleep 3000
 return FALSE
endif

if (ApplicErrorRetVal) then
 return FALSE
endif

Readlib "Deautl" EditTable, ToggleForm, EdFldView,
 HelpKey, EditCancel, SEditDoIt

opResult = EditTable("Apwtal", "Apwtal", "", "1", FALSE,
 "SEditDoIt", "",
 "",
 TRUE, FALSE, FALSE)

Release Procs EditTable, ToggleForm, EdFldView,
 HelpKey, EditCancel, SEditDoIt

return opResult
endproc

Writelib AppLib DealS3
Release Procs DealS3

proc DealS4()
private opResult

if (isempty("Apwtal")) then
 Message "No records to edit"
 Sleep 3000
 return FALSE
endif

if (ApplicErrorRetVal) then
 return FALSE
endif

```

Readlib "Deautl" EditTable, ToggleForm, EdFldView,
    HelpKey, EditCancel, SEditDoIt, SEditDelNoIns

opResult = EditTable("Apwtal", "Apwtal", "", "2", FALSE,
    "SEditDoIt", "SEditDelNoIns",
    "[Del] - Delete a record",
    TRUE, FALSE, FALSE)

Release Procs EditTable, ToggleForm, EdFldView,
    HelpKey, EditCancel, SEditDoIt, SEditDelNoIns

return opResult
endproc

Writelib AppLib DealS4
Release Procs DealS4

proc DealMenu()
private x, escape, zzzmexit, zzzzexit

    zzzzexit = FALSE
    x = "LIST Aliases"
    while (TRUE)
        Clear

        ShowMenu
            "LIST Aliases": "View a list of Aliases and
associated Data Elements.",
            "ADD Aliases": "Add an Alias and associate it with a
Data Element.",
            "EDIT Aliases": "Edit Aliases.",
            "DELETE Aliases": "Delete Aliases.",
            "PRINT Alias Reports": "Move to a lower menu for
Alias Report options.",
            "HELP": "Help with this menu.",
            "Leave": "Leave the application"
        Default x
        To x

    switch
        case x = "LIST Aliases":
            ReadLib "Deal" DealS1
            escape = DealS1()
            escape = not escape
            Release Procs DealS1

        case x = "ADD Aliases":

```

```

        ReadLib "Deal" DealS2
        escape = DealS2()
        escape = not escape
        Release Procs DealS2

    case x = "EDIT Aliases":
        ReadLib "Deal" DealS3
        escape = DealS3()
        escape = not escape
        Release Procs DealS3

    case x = "DELETE Aliases":
        ReadLib "Deal" DealS4
        escape = DealS4()
        escape = not escape
        Release Procs DealS4

    case x = "PRINT Alias Reports":
        Play "Deapt"
        x = "PRINT Alias Reports"
        escape = FALSE

    case x = "HELP":
        ReadLib "Deautl" PlayHelp

PlayHelp("Deahl")
Release Procs PlayHelp

        escape = FALSE

    case x = "Leave":
        ShowMenu
            "No": "Do not leave the application.",
            "Yes": "Leave the application."
        To zzzmexit

        zzzzexit = (zzzmexit = "Yes")
        escape = (zzzmexit = "Esc")

    case x = "Esc":
        escape = FALSE
endswitch

Reset
; reset ErrorProc value
ErrorProc = "ApplicErrorProc"
ApplicErrorRetVal = FALSE

```

```

        if (zzzzexit) then
            return TRUE
        endif

        if (not escape) then
            x = "LIST Aliases"
        endif
    endwhile
endproc

```

```

Writelib AppLib DealMenu
Release Procs DealMenu

```

```

;*****
; Deapt Script
; Paradox 3.0 PAL
; Data Element Alias Printing options main script.
; Jack Bacheller 21Jul90
;*****

```

```

Echo Off
Clear
Reset
Cursor Off

```

```

; set up the error proc for the application

```

```

ReadLib "Deaptut1" ApplicErrorProc
ErrorProc = "ApplicErrorProc"
ApplicErrorRetVal = FALSE

```

```

; Start the application

```

```

ReadLib "Deapt1" Deapt1Menu
Deapt1Menu()
Release Procs Deapt1Menu

```

```

Clearall

```

```
;*****\*****
; Deapt1 Script
; Paradox 3.0 PAL
; Data Element Alias printing script amenu operations.
; Jack Bacheller 21Jul90
;*****
```

```
AppLib = "Deapt1"
if (not isfile(AppLib + ".lib")) then
    Createlib AppLib
endif
```

```
proc Deapt1S1()
private opResult
```

```
    Readlib "Deaptut1" ReportTable
```

```
    opResult = ReportTable("Apwtal", "Apwtal", "2", "Printer",
    "")
```

```
    Release Procs ReportTable
```

```
    return opResult
endproc
```

```
WriteLib AppLib Deapt1S1
Release Procs Deapt1S1
```

```
proc Deapt1S2()
private opResult, tbl, rt, EscEnter, data
```

```
    Play "Deaptq4" ; put query on workspace
    if (ApplicErrorRetVal) then
        ClearAll
        return FALSE
    endif
```

```
    Readlib "Deaptut1" EnterVal
; get value for variable data
    data = EnterVal("Enter the Data Element Number. ",
    "N", "", 0)
    Release Procs EnterVal
    if (EscEnter) then
        ClearAll
        return FALSE
    endif
```

```

Readlib "Deaptut1" QueryDoIt
rt = QueryDoIt()
Release Procs QueryDoIt

if (not rt) then
    return FALSE
endif

Readlib "Deaptut1" ReportTable

opResult = ReportTable("Answer", "Apwtal", "1", "Printer",
"")

Release Procs ReportTable

return opResult
endproc

Writelib AppLib Deapt1S2
Release Procs Deapt1S2

proc Deapt1S3()
private opResult, tbl, rt, EscEnter, alias

    Play "Deaptq1" ; put query on workspace
    if (ApplicErrorRetVal) then
        ClearAll
        return FALSE
    endif

    Readlib "Deaptut1" EnterVal
; get value for variable alias
    alias = EnterVal("Enter the Alias Number. ", "N", "",
0)
    Release Procs EnterVal
    if (EscEnter) then
        ClearAll
        return FALSE
    endif

    Readlib "Deaptut1" QueryDoIt
    rt = QueryDoIt()
    Release Procs QueryDoIt

    if (not rt) then
        return FALSE
    endif

```

```

Readlib "Deaptut1" ReportTable

opResult = ReportTable("Answer", "Apwtal", "R", "Printer",
"")

Release Procs ReportTable

return opResult
endproc

Writelib AppLib Deapt1S3
Release Procs Deapt1S3

proc Deapt1S4()
private opResult, tbl, rt, EscEnter, ali

Play "Deaptq2" ; put query on workspace
if (ApplicErrorRetVal) then
ClearAll
return FALSE
endif

Readlib "Deaptut1" EnterVal
; get value for variable ali
ali = EnterVal("Enter the Alias Number. ", "N", "", 0)
Release Procs EnterVal
if (EscEnter) then
ClearAll
return FALSE
endif

Readlib "Deaptut1" QueryDoIt
rt = QueryDoIt()
Release Procs QueryDoIt

if (not rt) then
return FALSE
endif

Readlib "Deaptut1" ReportTable

opResult = ReportTable("Answer", "Apwtaldi", "R",
"Printer", "")

Release Procs ReportTable

return opResult

```

endproc

Writelib AppLib Deapt1S4
Release Procs Deapt1S4

proc Deapt1S5()
private opResult, tbl, rt, EscEnter, aliad

 Play "Deaptq3" ; put query on workspace
 if (ApplicErrorRetVal) then
 ClearAll
 return FALSE
 endif

 Readlib "Deaptut1" EnterVal
 ; get value for variable aliad
 aliad = EnterVal("Enter the Alias Number. ", "N", "",
0)
 Release Procs EnterVal
 if (EscEnter) then
 ClearAll
 return FALSE
 endif

 Readlib "Deaptut1" QueryDoIt
 rt = QueryDoIt()
 Release Procs QueryDoIt

 if (not rt) then
 return FALSE
 endif

 Readlib "Deaptut1" ReportTable

 opResult = ReportTable("Answer", "Apwtalqd", "R",
"Printer", "")

 Release Procs ReportTable

 return opResult
endproc

Writelib AppLib Deapt1S5
Release Procs Deapt1S5

proc Deapt1Menu()
private x, escape, zzzmexit, zzzzexit

```
zzzzexit = FALSE
x = "PRINT Alias List"
while (TRUE)
  Clear
```

```
  ShowMenu
```

```
    "PRINT Alias List": "Print a list of Aliases and
associated Data Element Numbers.",
    "PRINT Data Elmnt Als": "Print a list of Aliases
based on the input of Data Elmnt #.",
    "PRINT Alias Report": "Print a Report on a specific,
selected Alias.",
    "PRINT Qual Values": "Print the Data Values for a
Qualitative Alias.",
    "PRINT Quan Values": "Print the Data Values for a
Quantitative Alias.",
    "Leave": "Leave the Alias Print application"
  Default x
  To x
```

```
switch
```

```
  case x = "PRINT Alias List":
    ReadLib "Deapt1" Deapt1S1
    escape = Deapt1S1()
    escape = not escape
    Release Procs Deapt1S1

  case x = "PRINT Data Elmnt Als":
    ReadLib "Deapt1" Deapt1S2
    escape = Deapt1S2()
    escape = not escape
    Release Procs Deapt1S2

  case x = "PRINT Alias Report":
    ReadLib "Deapt1" Deapt1S3
    escape = Deapt1S3()
    escape = not escape
    Release Procs Deapt1S3

  case x = "PRINT Qual Values":
    ReadLib "Deapt1" Deapt1S4
    escape = Deapt1S4()
    escape = not escape
    Release Procs Deapt1S4

  case x = "PRINT Quan Values":
    ReadLib "Deapt1" Deapt1S5
    escape = Deapt1S5()
```

```

        escape = not escape
        Release Procs Deapt1S5

        case x = "Leave":
            ShowMenu
                "No": "Do not leave the application.",
                "Yes": "Leave the application."
            To zzzmexit

            zzzzexit = (zzzmexit = "Yes")
            escape = (zzzmexit = "Esc")

            case x = "Esc":
                escape = FALSE
            endswitch

        Reset
        ; reset ErrorProc value
        ErrorProc = "ApplicErrorProc"
        ApplicErrorRetVal = FALSE

        if (zzzzexit) then
            return TRUE
        endif

        if (not escape) then
            x = "PRINT Alias List"
        endif
    endwhile
endproc

Writelib AppLib Deapt1Menu
Release Procs Deapt1Menu

```

```

;*****
; Cwd Script
; Paradox 3.0 PAL
; Class Word main script
; Jack Bacheller 21Jul90
;*****

Echo Off
Clear
Reset
Cursor Off

; set up the error proc for the application

ReadLib "Cwdut1" ApplicErrorProc
ErrorProc = "ApplicErrorProc"
ApplicErrorRetVal = FALSE

; Start the application

ReadLib "Cwd1" Cwd1Menu
Cwd1Menu()
Release Procs Cwd1Menu

Clearall

;*****
; Cwd1 Script
; Paradox 3.0 PAL
; Class Word main menu operations.
; Jack Bacheller 21Jul90
;*****

AppLib = "Cwd1"
if (not isfile(AppLib + ".lib")) then
    Createlib AppLib
endif

proc Cwd1S1()
private opResult

    Readlib "Cwdut1" ViewTable, ToggleForm, VwFldView,
        HelpKey

```

```

opResult = ViewTable("Apwtcwor", "Apwtcwor", "3", FALSE)

Release Procs ViewTable, ToggleForm, VwFldView,
    HelpKey

    return opResult
endproc

Writelib AppLib Cwd1S1
Release Procs Cwd1S1

proc Cwd1S2()
private opResult

    Readlib "Cwdutl" EntryTable, KECheck, ToggleForm,
        EdFldView, HelpKey, EntryCancel, EntryDoIt,
        RenamePre, RenameSet, SaveList, CreateList,
        PrintList

    opResult = EntryTable("Apwtcwor", "", "F", FALSE)

    Release Procs EntryTable, KECheck, ToggleForm,
        EdFldView, HelpKey, EntryCancel, EntryDoIt,
        RenamePre, RenameSet, SaveList, CreateList,
        PrintList

    return opResult
endproc

Writelib AppLib Cwd1S2
Release Procs Cwd1S2

proc Cwd1S3()
private opResult

    if (isempty("Apwtcwor")) then
        Message "No records to edit"
        Sleep 3000
        return FALSE
    endif

    if (ApplicErrorRetVal) then
        return FALSE
    endif

    Readlib "Cwdutl" EditTable, ToggleForm, EdFldView,
        HelpKey, EditCancel, SEditDoIt, SEditDelNoIns

```

```

    opResult = EditTable("Apwtcwor", "Apwtcwor", "", "1",
FALSE,
    "SEditDoIt", "SEditDelNoIns",
    "[Del] - Delete a record",
    TRUE, FALSE, FALSE)

```

```

    Release Procs EditTable, ToggleForm, EdFldView,
        HelpKey, EditCancel, SEditDoIt, SEditDelNoIns

```

```

    return opResult
endproc

```

```

Writelib AppLib Cwd1S3
Release Procs Cwd1S3

```

```

proc Cwd1Menu()
private x, escape, zzzmexit, zzzzexit

```

```

    zzzzexit = FALSE
    x = "LIST Class Words"
    while (TRUE)
        Clear

```

```

        ShowMenu
            "LIST Class Words": "View a list of Class Words.",
            "ADD Class Words": "Add Class Words.",
            "EDIT/DELETE Class WD": "Edit or delete Class
Words.",
            "PRINT Class Words": "Move to a lower menu for Class
Word print options.",
            "Leave": "Leave the application"
        Default x
        To x

```

```

    switch
        case x = "LIST Class Words":
            ReadLib "Cwd1" Cwd1S1
            escape = Cwd1S1()
            escape = not escape
            Release Procs Cwd1S1

        case x = "ADD Class Words":
            ReadLib "Cwd1" Cwd1S2
            escape = Cwd1S2()
            escape = not escape
            Release Procs Cwd1S2

```

```

case x = "EDIT/DELETE Class WD":
    ReadLib "Cwd1" Cwd1S3
    escape = Cwd1S3()
    escape = not escape
    Release Procs Cwd1S3

case x = "PRINT Class Words":
    Play "Cwdpt"
    x = "PRINT Class Words"
    escape = FALSE

case x = "Leave":
    ShowMenu
        "No": "Do not leave the application.",
        "Yes": "Leave the application."
    To zzzmexit

    zzzzexit = (zzzmexit = "Yes")
    escape = (zzzmexit = "Esc")

case x = "Esc":
    escape = FALSE
endswitch

Reset
; reset ErrorProc value
ErrorProc = "ApplicErrorProc"
ApplicErrorRetVal = FALSE

if (zzzzexit) then
    return TRUE
endif

if (not escape) then
    x = "LIST Class Words"
endif
endwhile
endproc

Writelib AppLib Cwd1Menu
Release Procs Cwd1Menu

```

```
;*****  
; Cwdpt Script  
; Paradox 3.0 PAL  
; Class Word Printing main menu script.  
; Jack Bacheller 21Jul90  
;*****
```

```
Echo Off  
Clear  
Reset  
Cursor Off
```

```
; set up the error proc for the application
```

```
ReadLib "Cwdptut1" ApplicErrorProc  
ErrorProc = "ApplicErrorProc"  
ApplicErrorRetVal = FALSE
```

```
; Start the application
```

```
ReadLib "Cwdpt1" Cwdpt1Menu  
Cwdpt1Menu()  
Release Procs Cwdpt1Menu
```

```
Clearall
```

```

;*****
; Cwdpt1 Script
; Paradox 3.0 PAL
; Class Word Printing main menu operations.
; Jack Bacheller 21Jul90
;*****

```

```

AppLib = "Cwdpt1"
if (not isfile(AppLib + ".lib")) then
    Createlib AppLib
endif

```

```

proc Cwdpt1S1()
private opResult

```

```

    Readlib "Cwdptut1" ReportTable

```

```

    opResult = ReportTable("Apwtcwor", "Apwtcwor", "R",
"Printer", "")

```

```

    Release Procs ReportTable

```

```

    return opResult
endproc

```

```

Writelib AppLib Cwdpt1S1
Release Procs Cwdpt1S1

```

```

proc Cwdpt1S2()
private opResult, tbl, rt, EscEnter, name

```

```

    Play "Cwdptq1" ; put query on workspace
    if (ApplicErrorRetVal) then
        ClearAll
        return FALSE
    endif

```

```

    Readlib "Cwdptut1" EnterVal
; get value for variable name
    name = EnterVal("Enter the Class Word Name. ", "A20",
"", 0)
    Release Procs EnterVal
    if (EscEnter) then
        ClearAll
        return FALSE
    endif

```

```

Readlib "Cwdptut1" QueryDoIt
rt = QueryDoIt()
Release Procs QueryDoIt

if (not rt) then
    return FALSE
endif

Readlib "Cwdptut1" ReportTable

opResult = ReportTable("Answer", "Apwtcwor", "1",
"Printer", "")

Release Procs ReportTable

return opResult
endproc

Writelib AppLib Cwdpt1S2
Release Procs Cwdpt1S2

proc Cwdpt1Menu()
private x, escape, zzzmexit, zzzzexit

    zzzzexit = FALSE
    x = "PRINT Class Word Lst"
    while (TRUE)
        Clear

        ShowMenu
            "PRINT Class Word Lst": "Print a List of all of the
Class Words.",
            "PRINT CLASS Word": "Print a detailed report on a
specific, selected Class Word.",
            "Leave": "Leave the application"
        Default x
        To x

        switch
            case x = "PRINT Class Word Lst":
                ReadLib "Cwdpt1" Cwdpt1S1
                escape = Cwdpt1S1()
                escape = not escape
                Release Procs Cwdpt1S1

            case x = "PRINT CLASS Word":
                ReadLib "Cwdpt1" Cwdpt1S2
                escape = Cwdpt1S2()

```

```

        escape = not escape
        Release Procs Cwdpt1S2

    case x = "Leave":
        ShowMenu
            "No": "Do not leave the application.",
            "Yes": "Leave the application."
        To zzzmexit

        zzzzexit = (zzzmexit = "Yes")
        escape = (zzzmexit = "Esc")

    case x = "Esc":
        escape = FALSE
    endswitch

    Reset
    ; reset ErrorProc value
    ErrorProc = "ApplicErrorProc"
    ApplicErrorRetVal = FALSE

    if (zzzzexit) then
        return TRUE
    endif

    if (not escape) then
        x = "PRINT Class Word Lst"
    endif
endwhile
endproc

Writelib AppLib Cwdpt1Menu
Release Procs Cwdpt1Menu

```

```

;*****
; Pwd Script
; Paradox 3.0 PAL
; Prime Word Main menu script.
; Jack Bacheller 21Jul90
;*****

```

```

Echo Off
Clear
Reset
Cursor Off

```

```

; set up the error proc for the application

```

```

ReadLib "Pwduit1" ApplicErrorProc
ErrorProc = "ApplicErrorProc"
ApplicErrorRetVal = FALSE

```

```

; Start the application

```

```

ReadLib "Pwd1" Pwd1Menu
Pwd1Menu()
Release Procs Pwd1Menu

```

```

Clearall

```

```
;*****
; Pwd1 Script
; Paradox 3.0 PAL
; Prime Word main menu operations. Printing is included.
; Jack Bacheller 21Jul90
;*****
```

```
AppLib = "Pwd1"
if (not isfile(AppLib + ".lib")) then
    Createlib AppLib
endif
```

```
proc Pwd1S1()
private opResult

    Readlib "Pwdu1" ViewTable, ToggleForm, VwFldView,
        HelpKey

    opResult = ViewTable("Apwtpwor", "Apwtpwor", "1", FALSE)

    Release Procs ViewTable, ToggleForm, VwFldView,
        HelpKey

    return opResult
endproc
```

```
Writelib AppLib Pwd1S1
Release Procs Pwd1S1
```

```
proc Pwd1S2()
private opResult

    Readlib "Pwdu1" EntryTable, KECheck, ToggleForm,
        EdFldView, HelpKey, EntryCancel, EntryDoIt,
        RenamePre, RenameSet, SaveList, CreateList,
        PrintList

    opResult = EntryTable("Apwtpwor", "", "F", FALSE)

    Release Procs EntryTable, KECheck, ToggleForm,
        EdFldView, HelpKey, EntryCancel, EntryDoIt,
        RenamePre, RenameSet, SaveList, CreateList,
        PrintList

    return opResult
endproc
```

```
Writelib AppLib Pwd1S2
Release Procs Pwd1S2
```

```
proc Pwd1S3()
private opResult
```

```
    if (isempty("Apwtpwor")) then
        Message "No records to edit"
        Sleep 3000
        return FALSE
    endif
```

```
    if (ApplicErrorRetVal) then
        return FALSE
    endif
```

```
    Readlib "Pwduit1" EditTable, ToggleForm, EdFldView,
        HelpKey, EditCancel, SEditDoIt, SEditDelNoIns
```

```
    opResult = EditTable("Apwtpwor", "Apwtpwor", "", "2",
FALSE,
        "SEditDoIt", "SEditDelNoIns",
        "[Del] - Delete a record",
        TRUE, FALSE, FALSE)
```

```
    Release Procs EditTable, ToggleForm, EdFldView,
        HelpKey, EditCancel, SEditDoIt, SEditDelNoIns
```

```
    return opResult
endproc
```

```
Writelib AppLib Pwd1S3
Release Procs Pwd1S3
```

```
proc Pwd1S4()
private opResult
```

```
    Readlib "Pwduit1" ReportTable
```

```
    opResult = ReportTable("Apwtpwor", "Apwtpwor", "R",
"Printer", "")
```

```
    Release Procs ReportTable
```

```
    return opResult
endproc
```

```
Writelib AppLib Pwd1S4
Release Procs Pwd1S4
```

```
proc Pwd1Menu()
private x, escape, zzzmexit, zzzzexit
```

```
    zzzzexit = FALSE
    x = "LIST Prime Words"
    while (TRUE)
        Clear
```

```
    ShowMenu
```

```
        "LIST Prime Words": "View a list of Prime Words.",
        "ADD Prime Words": "Add Prime Words.",
        "EDIT/DELETE Prime WD": "Edit or delete Prime
words.",
        "PRINT Prime Words": "Print a list of Prime Words.",
        "Leave": "Leave the application"
    Default x
    To x
```

```
switch
```

```
    case x = "LIST Prime Words":
        ReadLib "Pwd1" Pwd1S1
        escape = Pwd1S1()
        escape = not escape
        Release Procs Pwd1S1
```

```
    case x = "ADD Prime Words":
        ReadLib "Pwd1" Pwd1S2
        escape = Pwd1S2()
        escape = not escape
        Release Procs Pwd1S2
```

```
    case x = "EDIT/DELETE Prime WD":
        ReadLib "Pwd1" Pwd1S3
        escape = Pwd1S3()
        escape = not escape
        Release Procs Pwd1S3
```

```
    case x = "PRINT Prime Words":
        ReadLib "Pwd1" Pwd1S4
        escape = Pwd1S4()
        escape = not escape
        Release Procs Pwd1S4
```

```
    case x = "Leave":
        ShowMenu
```

```

        "No": "Do not leave the application.",
        "Yes": "Leave the application."
    To zzzmexit

    zzzzexit = (zzzmexit = "Yes")
    escape = (zzzmexit = "Esc")

    case x = "Esc":
        escape = FALSE
    endswitch

    Reset
    ; reset ErrorProc value
    ErrorProc = "ApplicErrorProc"
    ApplicErrorRetVal = FALSE

    if (zzzzexit) then
        return TRUE
    endif

    if (not escape) then
        x = "LIST Prime Words"
    endif
endwhile
endproc

Writelib AppLib PwdlMenu
Release Procs PwdlMenu

```

LIST OF REFERENCES

1. The Deputy Secretary of Defense Memorandum dtd 4 Oct 1989, DOD Corporate Information Management.
2. FY 1991 Budget Review, Data Management Report Decision 925, 10 Oct 1989.
3. Army Regulation 25-9, Army Data Management and Standards Program, 25 Sep 1989.
4. Leong-Hong, B., Plagman, B., Data Dictionary/Directory Systems Administration, Implementation and Usage, John Wiley & Sons, 1982.
5. ADMS 25-VA4-APW-IBM-EM, End-User Manual for the Army Data Dictionary (ADD) Automated Dictionary Support System, 11 April 1990.
6. Kroenke, D., Dolan, K., Database Processing, Fundamentals, Design, Implementation, 3d ed., pp. 14-16, Science Research Associates Inc., 1988.
7. Interview between Phillip Olson, Lieutenant Colonel, USA, Corporate Information Management (DOD), Washington, DC, and the author 30 April, 1990.
8. Army Regulation 25-9, Army Data Management and Standards Program, Appendix G, 25 Sep 1989.
9. IDA Document D-629 ATCCIS Working Paper 7L, Operational and Procedural Requirements for Data Management and Standardization, Edition 1, Institute for Defense Analyses, June 1989.
10. Kroenke, D., Dolan, K., Database Processing, Fundamentals, Design, Implementation, 3d ed., pp. 90-97, Science Research Associates Inc., 1988.
11. Yourdon, E., Modern Structured Analysis, Yourdon Press, 1988.

12. Kroenke, D., Dolan, K., Database Processing, Fundamentals, Design, Implementation, 3d ed., p. 131, Science Research Associates Inc., 1988.
13. Kamel, M., "IS 4183 Lecture Notes: Applications of Database Management Systems," Naval Postgraduate School, December 1989.
14. Petreley, N., "Analyzing Relational Databases," Infoworld, pp. 51-68, 8 Jan 1990.
15. Borland Corporation, Paradox 3.0 User's Guide, 1989.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
Cameron Station
Alexandria, Virginia 22304-6145
2. Library, Code 52 2
Naval Postgraduate School
Monterey, California 93943-5000
3. Commandant of the Marine Corps 1
Code TE 06
Headquarters, U.S. Marine Corps
Washington, D.C. 20380-0001
4. Computer Technology Programs, Code 37 1
Naval Postgraduate School
Monterey, California 93943-5000
5. Professor Daniel R. Dolk, Code ASDK 1
Naval Postgraduate School
Monterey, California 93943-5000
6. Professor Magdi N. Kamel, Code 54KA 1
Naval Postgraduate School
Monterey, California 93943-5000
7. Captain John S. Bacheller USMC 2
409 Hilary Dr.
Tiburon, California 94920
8. HQDA ODISC4 1
Interoperability and Standards Office
ATTN: SAIS-ADO (COL. R. Potts)
The Pentagon, Room 1C670
Washington, D.C. 20310-0107
9. Corporate Information Management (CIM) 1
ATTN: LTCOL. P. Olson
The Pentagon, Room 1C535
Washington, D.C. 20310